International Journal of Control

ISSN: 0020-7179 (Print) 1366-5820 (Online) Journal homepage: http://www.tandfonline.com/loi/tcon20

Decentralised robust flow controller design for networks with multiple bottlenecks

İnci Munyas , Özen Yelbaşi , Enis Biberović , Altuğ İftar & Hitay Özbay

To cite this article: İnci Munyas , Özen Yelbaşi , Enis Biberović , Altuğ İftar & Hitay Özbay (2009) Decentralised robust flow controller design for networks with multiple bottlenecks, International Journal of Control, 82:1, 95-116

To link to this article: http://dx.doi.org/10.1080/00207170801993561

Published online: 21 Oct 2008.

International Journal of Control
Vol. 82, No. 1, January 2009, 95–116

Taylor & Francis
Taylor & Francis Group

Decentralised robust flow controller design for networks with multiple bottlenecks

İnci Munyas[ad], Özen Yelbaşi[b], Enis Biberović[cd], Altuğ İftar[b]* and Hitay Özbay[e]

[a]*Tusaş Aerospace Industries Inc., Ankara, Turkey;* [b]*Department of Electrical and Electronics Engineering, Anadolu University, Eskişehir, Turkey;* [c]*HERMES SoftLab d.o.o., Sarajevo, Trg Solidarnosti 2, Sarajevo, Bosnia and Herzegovina;* [d]*Department of Electrical and Electronics Engineering, Anadolu University;* [e]*Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey*

(*Received 27 December 2005; final version received 16 February 2008*)

Decentralised rate-based flow controller design in multi-bottleneck data-communication networks is considered. An $\mathcal{H}^\infty$ problem is formulated to find decentralised controllers which can be implemented locally at the bottleneck nodes. A suboptimal solution to this problem is found and the implementation of the decentralised controllers is presented. The controllers are robust to time-varying uncertain multiple time-delays in different channels. They also satisfy tracking and weighted fairness requirements. Lower bounds on the actual stability margins are derived and their relation to the design parameters is analysed. A number of simulations are also included to illustrate the time-domain performance of the proposed controllers.

Keywords: communication networks; flow control; robust control; decentralised control; time-delay systems; $\mathcal{H}^\infty$ control

1. Introduction

A modern communication network is expected to provide fast transmission with minimum loss. While guaranteeing the users such reliability, the resources of the network, such as buffers, bandwidth, etc., should be used efficiently. This resource management problem can be solved by controlling the traffic on the network; that is, using flow and congestion control mechanisms.

Congestion may cause long queueing delays and cell losses. It may be avoided by preventing the users from transmitting at rates faster than the rates allowed by the network. The congestion control mechanisms that use the rate at which the user should transmit as the feedback information are called rate-based (Bonomi and Fendick 1995) and the ones that use the window size, which is the number of packets that must be sent in a round trip time, as the feedback information are called window-based (Floyd 1994; Kung and Morris 1995; Kunniyur and Srikant 2000). Although window-based control is widely used for end to end congestion control in TCP/IP networks, rate-based control is preferred for edge to edge control in newer generation networks (Mascolo 2000; Laberteaux, Rohrs and Antsaklis 2002).

When the controller design for flow or congestion control mechanisms is considered, the main difficulty is that there exist relatively large transmission and propagation delays in high-speed networks (delay-bandwidth product is large). It should also be considered that these time-delays are usually uncertain and time-varying. Since there is usually more than one source connected to a bottleneck node, these time-delays are multiple. In the literature, there are many papers dealing with flow and congestion control in communication networks and many approaches to the flow controller design problem have been presented. In Altman, Başar and Srikant (1997), flow is controlled by the users and for the case of a team situation, a suboptimal control policy has been derived. In BenMohamed and Meerkov (1993), a congestion control algorithm is presented for single bottleneck networks and both adaptive and robust controllers are designed and some simulation results are given. The control algorithm in that work has been extended to the multiple bottleneck case in BenMohamed and Meerkov (1997). Other rate-based controller design approaches have been proposed in Ohsaki, Murata, Suzuki, Ikeda and Miyahara (1995a,b), Mascolo and Cavendish (1996), Floyd, Handley, Padhye and Widmer (2000), Mascolo (2000), Laberteaux, et al. (2002), Cavendish, Gerla and Mascolo (2004), among others.

In all the congestion controller design methods mentioned above, however, it is either assumed that there is no time-delay or that the time-delays are time-invariant. Time-varying uncertainties in the time-delays have explicitly been considered in

*Corresponding author. Email: aiftar@anadolu.edu.tr

ISSN 0020–7179 print/ISSN 1366–5820 online

DOI: 10.1080/00207170801993561
http://www.informaworld.com

Quet et al. (2002) and, using $\mathcal{H}^\infty$ control methods, a rate-based flow controller, robust to uncertain time-varying multiple time-delays in different channels, has been designed. However, in that work, only the single-bottleneck case has been considered. The multi-bottleneck case was considered in Biberović, İftar and Özbay (2001), where it was shown that decentralised flow controllers can be designed to solve the same problem in this case. The controller derivation, however, was not given in Biberović et al. (2001). The derivation of the controllers, for this case, has been shown and their implementation has been presented in Munyas, Yelbaşi and İftar (2003). Robustness of these controllers has been analysed in Munyas and İftar (2005a). In Biberović et al. (2001), Munyas et al. (2003) and Munyas and İftar (2005a) it was assumed that each bottleneck node acts as a virtual source for the next bottleneck node on the path of a connection. The case when only the data sending rates of the actual sources are controlled was later considered in Munyas and İftar (2005b).

In the present work, for the problem considered in Biberović et al. (2001) and Munyas et al. (2003), a parametrisation of the controllers to be implemented at the bottleneck nodes is given. Besides robustness, weighted fairness and tracking are also considered as design objectives. The design and implementation of the proposed controllers are demonstrated. Robustness of the controllers is also analysed using stability margins and a number of simulations are presented to show the time-domain performance of the proposed controllers in certain realistic cases. The actual contribution of the present work is in extending the results of Quet et al. (2002) to the multi-bottleneck case. To the authors' best knowledge, except for Biberović et al. (2001), Munyas et al. (2003) and Munyas and İftar (2005a,b), this is the first work which considers design of flow controllers which are robust to time-varying uncertainties in time-delays in the case of multiple bottleneck nodes.

Besides data-communication networks, the mathematical model considered in the present work appears in many other engineering applications, such as material transport systems (e.g. oil or gas pipelines, where simplified models of flow are used) and manufacturing systems, where continuous flow of parts to be processed can be seen as data flow. In this sense, the contribution of the present work is not restricted to data-communication networks. In fact, decentralised flow controller design approach presented here may be extended to any interconnected multivariable integrating system with time-delays, which may be uncertain and time-varying.

The organisation of this paper is as follows: in § 2, we consider the mathematical model of the multi-bottleneck system and the design problem of decentralised flow controllers; an $\mathcal{H}^\infty$ optimisation problem is considered in § 3, where the resulting decentralised controllers and their implementation are also presented; in § 4, the problem of fairly allocating the steady-state bandwidth to the users is considered and weighted fairness coefficients are obtained. The lower bounds for the actual stability margins for the uncertainties in the multiple time-delays and for the rate of change of the time-delays are derived in § 5 and their relation to the design parameters is analysed; § 6 contains a number of simulations that present the time-domain performance of the controllers; concluding remarks are made in the last section.

2. Problem statement

2.1 *Network model*

In this work, as in Biberović et al. (2001), we consider a network which consists of n bottleneck nodes and n_i sources directly (in the sense that there are no other bottlenecks on the path from that source to that bottleneck; there may however exist other nodes which are not bottlenecked) feeding the ith bottleneck node. Note that, if any physical source sends data to more than one bottleneck node, this source may be considered as a different source for each bottleneck node for the purpose of controller design. We also assume that, besides the sources, each bottleneck can also send data through other bottlenecks; i.e., each bottleneck is also a 'virtual source' for the next bottleneck on its path. Each bottleneck calculates not only the sending rates of its sources, but also the sending rates of the other bottlenecks which directly feed itself. Figure 1 shows the network for the case when there are two bottleneck nodes.

In a data-communication network, data packets are handled individually, and hence, data flow consists of discrete entities. For the purpose of controller design, however, we will use a continuous flow model. Such a model is often used by many researchers (e.g., see Chapters 5 and 6 of Srikant (2004) and references therein) and is usually named as a fluid-flow model. While running simulations in § 6, however, we will use a more realistic discrete model and show that a controller based on a fluid-flow model can also work well when the actual flow is discrete.

The dynamics of the queue length at the ith bottleneck node in our fluid-flow model are described as

$$\dot{q}_i(t) = \sum_{j=1}^{n_i} r^b_{i,j}(t) + \sum_{k=1,\, k\neq i}^{n} \rho^b_{k,i}(t) - c_i(t) - \sum_{k=1,\, k\neq i}^{n} \rho^s_{i,k}(t), \tag{1}$$

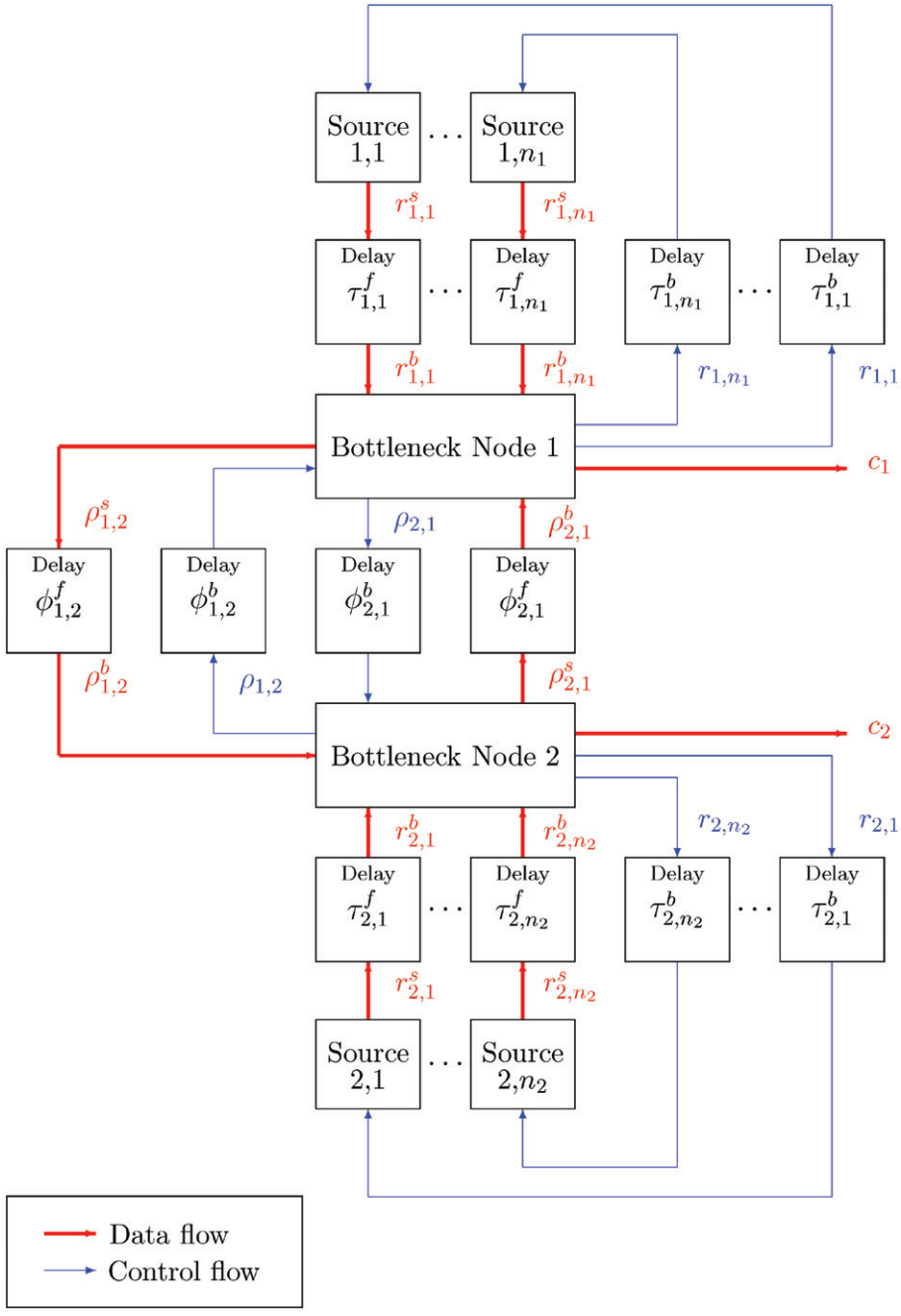

Figure 1. Network model for the two bottleneck node case.

where

$q_i(t)$ is the queue length at the ith bottleneck node at time t ($i=1,2,\ldots,n$),

$r^b_{i,j}(t)$ is the rate of data received at the ith bottleneck node from the jth source of the ith bottleneck node at time t ($i=1,2,\ldots,n$, $j=1,2,\ldots,n_i$),

$\rho^b_{k,i}(t)$ is the rate of data received at the ith bottleneck node at time t from the kth bottleneck node ($i=1,2,\ldots,n$, $k=1,2,\ldots,n, i\neq k$),

$c_i(t)$ is the outgoing flow rate, except for the flow going to the other bottleneck nodes, of the ith bottleneck node at time t ($i=1,2,\ldots,n$), and

$\rho^s_{i,k}(t)$ is the rate of data sent from the ith to the kth bottleneck node at time t ($i=1,2,\ldots,n$, $k=1,2,\ldots,n, i\neq k$).

The round-trip delay at time t for the flow from the jth source of the ith bottleneck node to the ith bottleneck node is given as

$$\tau_{i,j}(t) = \tau^b_{i,j}(t) + \tau^f_{i,j}(t) = h^r_{i,j} + \delta^r_{i,j}(t),$$

where $h^r_{i,j}$ is the time-invariant nominal part and $\delta^r_{i,j}(t)$ is the time-varying uncertain part. Similarly, the round-trip delay at time t for the flow from the ith to the kth bottleneck node is given as

$$\phi_{i,k}(t) = \phi^b_{i,k}(t) + \phi^f_{i,k}(t) = h^\rho_{i,k} + \delta^{\rho}_{i,k}(t),$$

where $h^\rho_{i,k}$ is the time-invariant nominal part and $\delta^\rho_{i,k}(t)$ is the time-varying uncertain part. In these terms,

$\tau^b_{i,j}(t) := h^{rb}_{i,j} + \delta^{rb}_{i,j}(t)$ represents the backward time-delay from the controller implemented at the ith bottleneck node to the jth source of the ith bottleneck node (the time-delay which occurs between the time a command signal for a rate is issued and the actual time this rate is set) where $h^{rb}_{i,j}$ is the nominal time-invariant known backward delay and $\delta^{rb}_{i,j}(t)$ is the time-varying backward time-delay uncertainty,

$\tau^f_{i,j}(t) := h^{rf}_{i,j} + \delta^{rf}_{i,j}(t)$ represents the forward time-delay from the jth source of the ith bottleneck node to the ith bottleneck node (the time-delay which is required for the data to reach the bottleneck node) where $h^{rf}_{i,j}$ is the nominal time-invariant known forward delay and $\delta^{rf}_{i,j}(t)$ is the time-varying forward time-delay uncertainty,

$\phi^b_{i,k}(t) := h^{\rho b}_{i,k} + \delta^{\rho b}_{i,k}(t)$ represents the backward time-delay from the controller at the kth bottleneck node to the ith bottleneck node where $h^{\rho b}_{i,k}$ is the nominal time-invariant known backward delay and $\delta^{\rho b}_{i,k}(t)$ is the time-varying backward time-delay uncertainty,

$\phi^f_{i,k}(t) := h^{\rho f}_{i,k} + \delta^{\rho f}_{i,k}(t)$ represents the forward time-delay from the ith bottleneck node to the kth bottleneck node where $h^{\rho f}_{i,k}$ is the nominal time-invariant known forward delay and $\delta^{\rho f}_{i,k}(t)$ is the time-varying forward time-delay uncertainty.

To determine $r^b_{i,j}(t)$ in (1), the total amount of data received at the ith bottleneck node from its jth source is written as follows (Quet et al. 2002):

$$\int_0^t r^b_{i,j}(\nu)\mathrm{d}\nu = \begin{cases} \int_0^{t-\tau^f_{i,j}(t)} r^s_{i,j}(\varphi)\mathrm{d}\varphi, & t-\tau^f_{i,j}(t) \geq 0 \\ 0, & t-\tau^f_{i,j}(t) < 0, \end{cases} \tag{2}$$

where

$r^s_{i,j}(t)$ is the rate of data sent from the jth source of the ith bottleneck node at time t ($i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$).

Similarly, to determine $\rho^b_{k,i}(t)$ in (1), the total amount of data received at the ith bottleneck node from the kth bottleneck node is written as,

$$\int_0^t \rho^b_{k,i}(\nu)\mathrm{d}\nu = \begin{cases} \int_0^{t-\phi^f_{k,i}(t)} \rho^s_{k,i}(\varphi)\mathrm{d}\varphi, & t-\phi^f_{k,i}(t) \geq 0 \\ 0, & t-\phi^f_{k,i}(t) < 0. \end{cases} \tag{3}$$

Since there is a time-varying backward time-delay, $\phi^b_{k,i}(t)$, between the ith and the kth bottleneck nodes, we have $\rho^s_{k,i}(t) = \rho_{k,i}(t - \phi^b_{k,i}(t))$, where

$\rho_{k,i}(t)$ is the flow rate command at time t for the flow from the kth to the ith bottleneck node ($i = 1, 2, \ldots, n$, $k = 1, 2, \ldots, n$, $i \neq k$), which must be computed (by the controller to be designed) at the ith bottleneck node.

Similarly, since there is a time-varying backward time-delay, $\tau^b_{i,j}(t)$, between the ith bottleneck node and its jth source, $r^s_{i,j}(t) = r_{i,j}(t - \tau^b_{i,j}(t))$, where

$r_{i,j}(t)$ is the flow rate command at time t for the flow from the jth source of the ith bottleneck node to the ith bottleneck node ($i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$), which must be computed (by the controller to be designed) at the ith bottleneck node.

Taking the derivatives of both sides of (2) and (3), the data receiving rates at the ith bottleneck node from its jth source, $r^b_{i,j}(t)$, and from the kth bottleneck node, $\rho^b_{k,i}(t)$, can be found as

$$r^b_{i,j}(t) = \begin{cases} (1 - \dot{\delta}^{rf}_{i,j}(t))r_{i,j}(t - \tau_{i,j}(t)), & t-\tau^f_{i,j}(t) \geq 0 \\ 0, & t-\tau^f_{i,j}(t) < 0, \end{cases} \tag{4}$$

and

$$\rho^b_{k,i}(t) = \begin{cases} (1-\dot{\delta}^{\rho f}_{k,i}(t))\rho_{k,i}(t-\phi_{k,i}(t)), & t-\phi^f_{k,i}(t) \geq 0 \\ 0, & t-\phi^f_{k,i}(t) < 0. \end{cases} \tag{5}$$

It is assumed that the uncertainties satisfy the following:

$$\left|\delta^r_{i,j}(t)\right| < \delta^{r+}_{i,j}, \quad \left|\dot{\delta}^r_{i,j}(t)\right| < \beta^r_{i,j}, \quad \left|\dot{\delta}^{rf}_{i,j}(t)\right| < \beta^{rf}_{i,j}, \tag{6}$$

$$\left|\delta^p_{i,k}(t)\right| < \delta^{\rho+}_{i,k}, \quad \left|\delta^{\rho b}_{i,k}(t)\right| < \delta^{\rho b+}_{i,k}, \quad \left|\dot{\delta}^{\rho}_{i,k}(t)\right| < \beta^{\rho}_{i,k}, \quad \left|\dot{\delta}^{\rho f}_{i,k}(t)\right| < \beta^{\rho f}_{i,k}, \quad \left|\dot{\delta}^{\rho b}_{i,k}(t)\right| < \beta^{\rho b}_{i,k}, \tag{7}$$

for all t, for some known bounds $\delta^{r+}_{i,j} > 0, 0 < \beta^{rf}_{i,j} < \beta^{r}_{i,j} < 1$, $0 < \delta^{\rho b+}_{i,k} < \delta^{\rho+}_{i,k}$, $0 < \beta^{\rho f}_{i,k}, \beta^{\rho b}_{i,k} < \beta^{\rho}_{i,k} < 1$ ($i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, n_i$, $k = 1, 2, \ldots, n$, $k \neq i$). It should be noted that, in a real application, there also exist some hard constraints, such as non-negativity constraints and upper bounds on the queue lengths and on the data rates. In this work, for the purpose of controller design, we will assume that these hard constraints are always satisfied. We will, however, consider such constraints in §6, while running simulations.

Remark 1: Besides the existence of mutiple bottleneck nodes (and hence multiple queues), the main difference between the model used here and in Quet et al. (2002) is the existence of flows between the bottleneck nodes (i.e. the terms $\rho^b_{k,i}$ and $\rho^s_{i,k}$ in (1)). These flows cause a coupling between the bottleneck nodes and must be explicitly considered in controller design as done in §3.

Remark 2: As mentioned in the introduction, the present model can also be used in other flow control problems, where flow can simply be modelled by, possibly time-varying and uncertain, time-delays. For example, in a gas transport system (where detailed modelling, e.g., using Navier-Stokes equations, is not found necessary, due to say almost constant pressure in a pipe) the two bottleneck nodes in Figure 1 can be considered as storage tanks. Forward delay lines would represent pipelines of different lengths; backward delays would indicate the communication delay between a local controller (implemented at the site of each storage tank) and the actuators (compressors implemented at the start of each pipeline feeding that storage tank) which adjust the flow rates. The sources, on the other hand, could be the supply reservoirs.

2.2 *Control problem*

The problem is to design decentralised controllers to be implemented at each bottleneck node, to regulate the queue length $q_i(t)$ at that node by determining the data sending rates of the sources and the other bottleneck nodes to that node. The desired queue length, $q_{d,i}$, at the ith bottleneck node is chosen to be some positive value (typically half of the buffer size) so that the outgoing link is not under-utilised.

As shown in Appendix A, the overall control system can be represented as in Figure 2. In this figure, K is the controller to be designed, P_o is the nominal plant, W_{21} and W_{22} are the weighting matrices, and Δ^o_{LTV} is an arbitrary linear time-varying system which represents the uncertainties. Exact expressions for $P_o(s)$, $W_{21}(s)$, and $W_{22}(s)$ are given in Appendix A. The structure of Δ^o_{LTV} is also given in Appendix A, and it is shown that the $\mathcal{L}_2$-induced norm of Δ^o_{LTV}, $\|\Delta^o_{LTV}\|$, is less than 1.

By using the small gain theorem (Zhou, Doyle and Glover 1995), the closed-loop system shown in Figure 2 is robustly stable for all $\|\Delta^o_{LTV}\| < 1$ if K stabilises P_o and

$$\left\| W_{22}K(I + P_oK)^{-1}W_{21} \right\|_\infty \leq 1 \tag{8}$$

is satisfied, where $\|\cdot\|_\infty$ denotes the $\mathcal{H}^\infty$ norm and I denotes the identity matrix. Using the fact that $W^T_{22}W_{22} = \hat{P}^T\hat{P} = I$ (see Appendix A for W_{22} and $\hat{P}$),

$$\left\| W_{22}K(I + P_oK)^{-1}W_{21} \right\|_\infty = \left\| \hat{P}K(I + P_oK)^{-1}W_{21} \right\|_\infty \tag{9}$$

is obtained. On the other hand, using the definition of W_{21} (see Appendix A), (9) can be bounded above by

$$\left\| \hat{P}K(I + P_oK)^{-1}W_{21} \right\|_\infty \leq \left\| \xi\hat{P}K(I + P_oK)^{-1} \right\|_\infty,$$

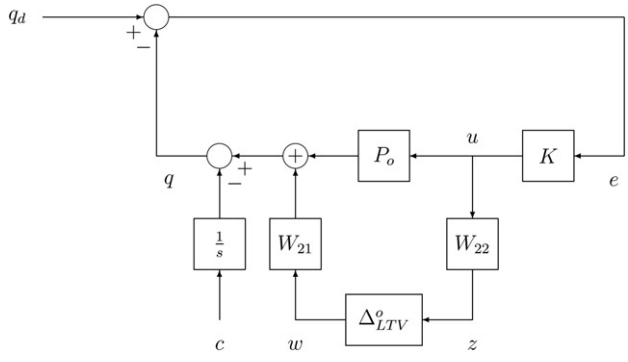

Figure 2. Overall control system.

where $\xi(s) := (1/s)\xi_1 + \xi_2$, with $\xi_1 := \max_i(\xi_{i,1})$ and $\xi_2 := \max_i(\xi_{i,2})$, where

$$\xi_{i,1} := \sqrt{\sum_{j=1}^{n_i}\left(e^r_{i,j,1}\right)^2 + 2\sum_{k=1,\,k\neq i}^{n}\left(e^\rho_{k,i,1}\right)^2 + 2\sum_{k=1,\,k\neq i}^{n}\left(e^{\rho b}_{i,k,1}\right)^2}, \tag{10}$$

$$\xi_{i,2} := \sqrt{\sum_{j=1}^{n_i}\left(e^r_{i,j,2}\right)^2 + 2\sum_{k=1,\,k\neq i}^{n}\left(e^\rho_{k,i,2}\right)^2 + 2\sum_{k=1,\,k\neq i}^{n}\left(e^{\rho b}_{i,k,2}\right)^2}, \tag{11}$$

where $e^r_{i,j,l}$, $e^\rho_{k,i,l}$, and $e^{\rho b}_{i,k,l}$ $(l = 1, 2)$ are parameters that depend on the bounds given in (6)–(7) and are defined in Appendix A. Thus, conservatively, (8) is satisfied if

$$\left\|\xi\hat{P}K(I + P_oK)^{-1}\right\|_\infty \leq 1. \tag{12}$$

Next, as in Quet et al. (2002), to guarantee tracking ($\lim_{t\to\infty} q_i(t) = q_{d,i}$) and good transient response, we formulate the problem

$$\text{minimise}\left\|W_1(I + P_oK)^{-1}\right\|_\infty \tag{13}$$

over all controllers K stabilising P_o, where $W_1(s) := (1/s^2)$.

Remark 3: Note that, Figure 2 resembles to Figure 2 in Quet et al. (2002). However, besides the fact that both P_o and K are multi-input multi-output in the present case (P_o is single-output and K is single-input in Quet et al. (2002)), the structures of Δ^o_{LTV} and W_{21} are different. Furthermore, a new block, W_{22}, is needed from u to z in the present case. These differences make the controller design more involved compared to Quet et al. (2002), as will be seen in the next section.

3. The $\mathcal{H}^\infty$ optimisation problem and controller design

Combining the robust stability, (12), and nominal performance, (13), conditions, we define the following two-block $\mathcal{H}^\infty$ optimisation problem:

$$\inf_{K \text{ stabilising } P_o}\left\|\begin{bmatrix} W_1(I + P_oK)^{-1} \\ \xi\hat{P}K(I + P_oK)^{-1}\end{bmatrix}\right\|_\infty =: \gamma^{\text{opt}}. \tag{14}$$

To find a solution to this problem, in Appendix B, following some transformations we decompose the problem into a number of subproblems, each of which involves a single delay. Then, using the results of Quet et al. (2002), and some transformations (see Appendix B), we obtain the following suboptimal controller to solve the optimisation problem (14):

$$K = \begin{bmatrix}\hat{K}^r \\ \sqrt{2}\hat{K}^\rho\end{bmatrix}, \tag{15}$$

where

$$\hat{K}^r = \begin{bmatrix} K^r_{11} & & \\ \vdots & 0 & \\ K^r_{1n_1} & & \\ & \ddots & \\ & & K^r_{n1} \\ & 0 & \vdots \\ & & K^r_{nn_n}\end{bmatrix} \quad \text{and}$$

$$\sqrt{2}\hat{K}^\rho = \begin{bmatrix} K^\rho_{21} & & \\ \vdots & 0 & \\ K^\rho_{n1} & & \\ & \ddots & \\ & & K^\rho_{1n} \\ & 0 & \vdots \\ & & K^\rho_{(n-1)n}\end{bmatrix},$$

where

$$K^r_{i,j} = \frac{C^r_{i,j}}{1 + C^r_{i,j}P^r_{i,j}}\left[1 - \sum_{k=1}^{n_i}\alpha^r_{i,k}\frac{C^r_{i,k}P^r_{i,k}}{1 + C^r_{i,k}P^r_{i,k}} - \sum_{k=1,k\neq i}^{n}\alpha^\rho_{k,i}\frac{C^\rho_{k,i}P^\rho_{k,i}}{1 + C^\rho_{k,i}P^\rho_{k,i}} - \sum_{k=1,k\neq i}^{n}\alpha^{\rho b}_{i,k}\frac{C^{\rho b}_{i,k}P^{\rho b}_{i,k}}{1 + C^{\rho b}_{i,k}P^{\rho b}_{i,k}}\right]^{-1}, \tag{16}$$

$$K^\rho_{j,i} = \frac{\sqrt{2}C^\rho_{j,i}}{1 + C^\rho_{j,i}P^\rho_{j,i}}\left[1 - \sum_{k=1}^{n_i}\alpha^r_{i,k}\frac{C^r_{i,k}P^r_{i,k}}{1 + C^r_{i,k}P^r_{i,k}} - \sum_{k=1,k\neq i}^{n}\alpha^\rho_{k,i}\frac{C^\rho_{k,i}P^\rho_{k,i}}{1 + C^\rho_{k,i}P^\rho_{k,i}} - \sum_{k=1,k\neq i}^{n}\alpha^{\rho b}_{i,k}\frac{C^{\rho b}_{i,k}P^{\rho b}_{i,k}}{1 + C^{\rho b}_{i,k}P^{\rho b}_{i,k}}\right]^{-1}. \tag{17}$$

Here, $P^r_{i,k}(s) := (1/\alpha^r_{i,k}s)e^{-h^r_{i,k}s}$, $P^\rho_{k,i}(s) := (\sqrt{2}/\alpha^\rho_{k,i}s)e^{-h^\rho_{k,i}s}$, and $P^{\rho b}_{i,k}(s) := (\sqrt{2}/\alpha^{\rho b}_{i,k}s)e^{-h^{\rho b}_{i,k}s}$ is the nominal plant for the subproblem with delay $h^r_{i,k}$, $h^\rho_{k,i}$, and $h^{\rho b}_{i,k}$, respectively. Furthermore, $C^\bullet_{i,k}$, is the optimal controller for the subproblem with the nominal plant $P^\bullet_{i,k}$, where superscript $\bullet$ represents r, ρ, or ρb, and is given by (42).

The design parameters $\alpha^{\bullet}_{i,k}$s are positive numbers satisfying

$$\sum_{l=1}^{n_i} \alpha^r_{i,l} + \sum_{l=1,\, l\neq i}^{n} \alpha^{\rho}_{l,\,i} + \sum_{l=1,\, l\neq i}^{n} \alpha^{\rho b}_{i,l} = 1 \tag{18}$$

for all $i=1,\ldots,n$. In the next section, we will show that these parameters can be used in allocating the steady-state bandwidth to the users fairly.

As seen from (15), the part of the controller for the ith bottleneck node gets feedback only from q_i to regulate the queue length q_i by determining the flow rates $r_{i,j}$, $j=1,\ldots,n_i$, and $\rho_{k,i}$, $k=1,\ldots,n, k\neq i$. Therefore, the controller is composed of n decentralised controllers:

$$K_i = \begin{bmatrix} \hat{K}^r_i \\ \sqrt{2}\hat{K}^{\rho}_i \end{bmatrix} = \begin{bmatrix} K^r_{i1} \\ \vdots \\ K^r_{in_i} \\ K^{\rho}_{1i} \\ \vdots \\ K^{\rho}_{(i-1)i} \\ K^{\rho}_{(i+1)i} \\ \vdots \\ K^{\rho}_{ni} \end{bmatrix}, \tag{19}$$

each of which can be implemented at the corresponding bottleneck node as shown in Figure 3. This controller stabilises the nominal plant and makes the $\mathcal{H}^{\infty}$ norm of the matrix in (14) less than some $\tilde{\gamma}$ (an upper bound that can be found from the γ's of the subproblems). Thus, as long as the hard constraints are satisfied, the controller stabilises the actual plant for all variations of the time-delays satisfying $|\delta^r_{i,j}(t)| < (\delta^{r+}_{i,j}/\tilde{\gamma})$, $|\dot{\delta}^r_{i,j}(t)| < (\beta^r_{i,j}/\tilde{\gamma})$, $|\delta^{rf}_{i,j}(t)| < (\beta^{rf}_{i,j}/\tilde{\gamma})$, $|\delta^{\rho}_{j,i}(t)| < (\delta^{\rho+}_{j,i}/\tilde{\gamma})$, $|\delta^{\rho b}_{i,j}(t)| < (\delta^{\rho b+}_{i,j}/\tilde{\gamma})$, $|\dot{\delta}^{\rho}_{j,i}(t)| < (\beta^{\rho}_{j,i}/\tilde{\gamma})$, $|\dot{\delta}^{\rho b}_{i,j}(t)| < (\beta^{\rho b}_{i,j}/\tilde{\gamma})$, and $|\dot{\delta}^{\rho f}_{j,i}(t)| < (\beta^{\rho f}_{j,i}/\tilde{\gamma})$. A more detailed analysis of stability margins in terms of the design parameters is given in § 5.

4. Weighted fairness

To maximise the network utilisation while satisfying the traffic contracts of the users, the bandwidth should be allocated to the users fairly. It may, however, be desired to assign different priorities to different sources and other bottleneck nodes which send data to a bottleneck node in the network. This can be done by allocating the available bandwidth of any particular bottleneck node to the users according to different weights at the steady-state. To see what these weights are, let us express the rate feedback signals as

$$u(s) = K(s)e(s), \tag{20}$$

where $u(s)$ is the Laplace transform of $u(t)$, which is given in (31) (with some abuse of notation, we will use the same symbol for a time signal and its Laplace transform), $e(s) = [\, e_1(s) \;\; \cdots \;\; e_n(s)\,]^T$ is the Laplace transform of $e(t) := q_d - q(t)$, and $q_d := [\, q_{d,1} \;\; \cdots \;\; q_{d,n}\,]^T$ is the vector of the desired queue lengths, which are assumed to be constant. Using the structure of the controller, given in (15), from (20) we obtain

$$r_{i,j}(s) = K^r_{i,j}(s)e_i(s), \quad j=1,\ldots,n_i, \tag{21}$$

and

$$\rho_{k,\,i}(s) = K^{\rho}_{k,\,i}(s)e_i(s), \quad k=1,\ldots,n, \quad k\neq i, \tag{22}$$

for $i=1,\ldots,n$. Using the queue length dynamics given in (1), the tracking error is obtained as follows:

$$\begin{aligned} e_i(s) = &-\frac{1}{s}\left(\sum_{j=1}^{n_i} r^b_{i,j}(s) + \sum_{k=1,\,k\neq i}^{n} \rho^b_{k,\,i}(s)\right) \\ &+ \frac{1}{s}\left(q_{d,\,i} + c_i(s) + \sum_{k=1,\,k\neq i}^{n} \rho^s_{i,\,k}(s)\right). \end{aligned} \tag{23}$$

It is known that $r^b_{i,j}(t)$ and $\rho^b_{k,\,i}(t)$ are respectively given by (4) and (5). For the nominal plant, we have $\delta^r_{i,j}(t) = \delta^{rf}_{i,j}(t) = \delta^{\rho}_{k,\,i}(t) = \delta^{\rho f}_{k,\,i}(t) = 0$. Hence, $r^b_{i,j}(t) = r_{i,j}(t - h^r_{i,j})$ and $\rho^b_{k,\,i}(t) = \rho_{k,\,i}(t - h^{\rho}_{k,\,i})$. Taking the Laplace transform of these expressions and substituting (21) and (22) into (23) lead to

$$\begin{aligned} e_i(s) = &\left(s + \sum_{j=1}^{n_i} e^{-h^r_{i,j}s} K^r_{i,j}(s) + \sum_{k=1,\,k\neq i}^{n} e^{-h^{\rho}_{k,\,i}s} K^{\rho}_{k,\,i}(s)\right)^{-1} \\ &\times \left(q_{d,\,i} + c_i(s) + \sum_{k=1,\,k\neq i}^{n} e^{-h^{\rho b}_{i,k}s} \rho_{i,\,k}(s)\right). \end{aligned}$$

Therefore, using this expression, together with (16), (17), and (42), in (21) and (22), the steady-state values of the rate feedback signals, $\lim_{t\to\infty} r_{i,j}(t)$ and $\lim_{t\to\infty}\rho_{k,i}(t)$, can be found as

$$\lim_{s\to 0} s r_{i,j}(s) = \frac{\alpha^r_{i,j}}{\tilde{\alpha}_i}\left(c_{i,\infty} + \sum_{l=1,\, l\neq i}^{n} \rho^{\infty}_{i,l}\right) \tag{24}$$

and

$$\lim_{s\to 0} s \rho_{k,\,i}(s) = \frac{\alpha^{\rho}_{k,\,i}}{\tilde{\alpha}_i}\left(c_{i,\infty} + \sum_{l=1,\, l\neq i}^{n} \rho^{\infty}_{i,l}\right) \tag{25}$$

respectively. Here, $\tilde{\alpha}_i := \sum_{j=1}^{n_i} \alpha^r_{i,j} + \sum_{k=1,\,k\neq i}^{n} \alpha^{\rho}_{k,\,i}$, $c_{i,\infty} := \lim_{t\to\infty}\; c_i(t) = \lim_{s\to 0}\; s c_i(s)$, and $\rho^{\infty}_{i,l} := \lim_{t\to\infty} \rho_{i,\,l}(t) = \lim_{s\to 0} s\rho_{i,\,l}(s)$. In this way, the available

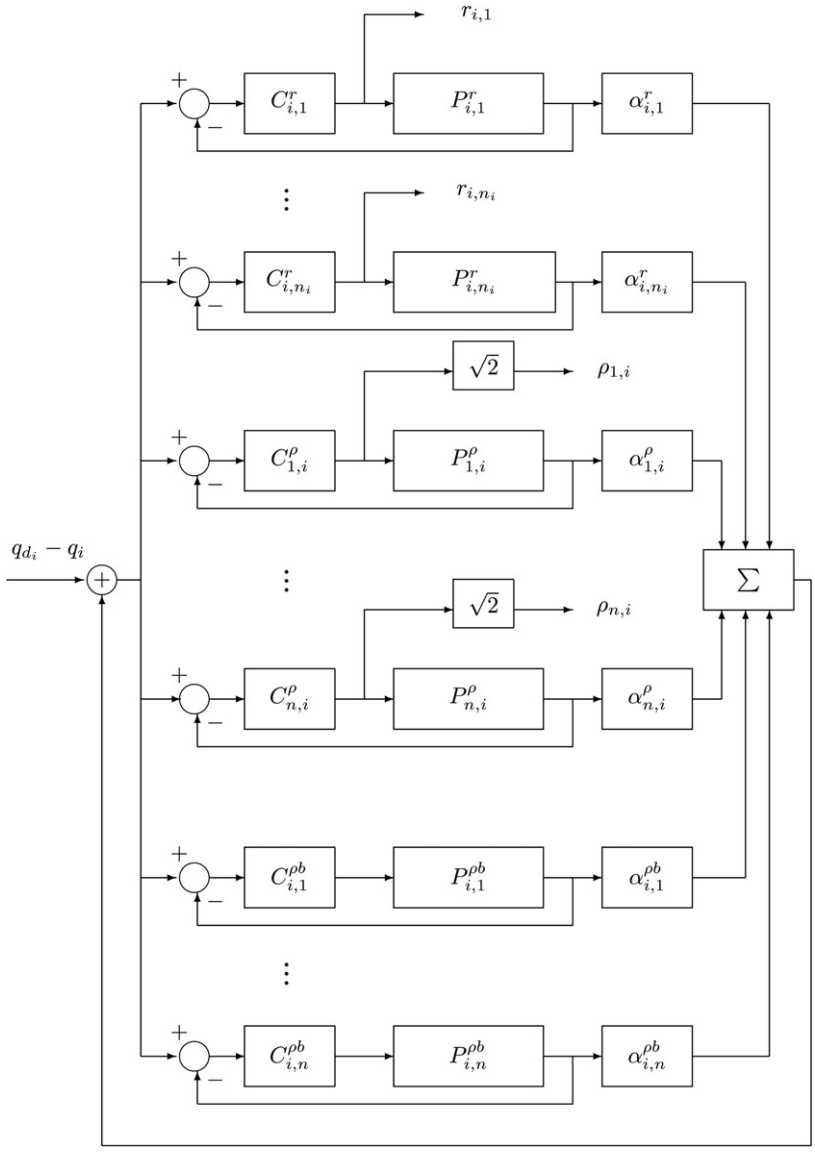

Figure 3. Implementation of the controller K_i.

bandwidth at the ith bottleneck node can be allocated to the users by using the design parameters $\alpha^r_{i,j}$s and $\alpha^o_{k,i}$s. Therefore, as in the single bottleneck case Quet, et al. (2002), these parameters can be regarded as fairness weights.

The steady-state values of the rate feedback signals can also be obtained in terms of $c_{i,\infty}$s alone, as shown in Appendix C.

5. Stability margins

For the closed-loop system shown in Figure 2 to be robustly stable for all $\|\Delta^o_{LTV}\| < 1$, K should stabilise P_o and (8) should be satisfied. Let $\bar{W} := \mathrm{diag}(\xi_1, \ldots, \xi_n)$, where $\xi_i(s) := (1/s)\xi_{i,1} + \xi_{i,2}$ with $\xi_{i,1}$ and $\xi_{i,2}$ are as given in (10) and (11), respectively, for all $i = 1, \ldots, n$. Then, using $W_{22}^T W_{22} = \hat{P}^T \hat{P} = I$ and $W_{21} W_{21}^* = \bar{W}\bar{W}^*$, it can be shown that (8) and

$$\left\| \hat{P} K (I + P_o K)^{-1} \bar{W} \right\|_\infty \leq 1 \tag{26}$$

are equivalent. Thus, if the following inequalities are satisfied, robust stability of the system is guaranteed (see Munyas and İftar (2005a), for details):

$$\left\{ \sum_{j=1}^{n_i} \left(e^{r,act}_{i,j,1} \right)^2 + \sum_{k=1,k\neq i}^{n} \left(e^{\rho,act}_{k,i,1} \right)^2 + \sum_{k=1,k\neq i}^{n} \left(e^{\rho b,act}_{i,k,1} \right)^2 \right\}$$
$$\leq \frac{1}{\tilde{\gamma}_i^2} \left\{ \sum_{j=1}^{n_i} \left(e^{r}_{i,j,1} \right)^2 + \sum_{k=1,k\neq i}^{n} \left(e^{\rho}_{k,i,1} \right)^2 + \sum_{k=1,k\neq i}^{n} \left(e^{\rho b}_{i,k,1} \right)^2 \right\} \tag{27}$$

and

$$\left\{\sum_{j=1}^{n_i}\left(e_{i,j,2}^{r,act}\right)^2+\sum_{k=1,k\neq i}^{n}\left(e_{k,i,2}^{\rho,act}\right)^2+\sum_{k=1,k\neq i}^{n}\left(e_{i,k,2}^{\rho b,act}\right)^2\right\}$$
$$\leq\frac{1}{\tilde{\gamma}_i^2}\left\{\sum_{j=1}^{n_i}\left(e_{i,j,2}^{r}\right)^2+\sum_{k=1,k\neq i}^{n}\left(e_{k,i,2}^{\rho}\right)^2+\sum_{k=1,k\neq i}^{n}\left(e_{i,k,2}^{\rho b}\right)^2\right\} \tag{28}$$

for $i=1,\ldots,n$, where $\tilde{\gamma}_i$ is as given in (40). Here, the actual stability margin for $e_{i,k,l}^{\bullet}$ is denoted by $e_{i,k,l}^{\bullet,act}$, where the superscript $\bullet$ represents r, ρ, or ρb. It is seen that the lower bounds for the actual stability margins for each bottleneck node can be calculated independently from the other bottleneck nodes. Since the number of sources and the number of other bottleneck nodes connected to a bottleneck node may be greater than 1, the inequalities in (27) and (28) lead to infinitely many solutions for the lower bounds and any one of the solutions will provide robust stability of the system.

To observe the effects of the uncertainty bounds used in the controller design, the lower bounds on the actual stability margins satisfying (27) and (28) are depicted for a number of example cases. To do this, first, the following terms are defined:

$$e_{i,l}^{r,act}:=\sqrt{\sum_{j=1}^{n_i}\left(e_{i,j,l}^{r,act}\right)^2},\quad e_{i,l}^{\rho,act}:=\sqrt{\sum_{k=1,k\neq i}^{n}\left(e_{k,i,l}^{\rho,act}\right)^2},$$
$$e_{i,l}^{\rho b,act}:=\sqrt{\sum_{k=1,k\neq i}^{n}\left(e_{i,k,l}^{\rho b,act}\right)^2},$$

where $i=1,\ldots,n$ and $l=1,\ 2$. Here, $e_{i,1}^{r,act}$ gives a measure for the actual stability margin relating to the rate of change of $\delta_{i,j}^{r}(t)$, $e_{i,1}^{\rho,act}$ gives a measure for the actual stability margin relating to the rate of change of $\delta_{k,i}^{\rho}(t)$, and $e_{i,1}^{\rho b,act}$ gives a measure for the actual stability margin relating to the rate of change of $\delta_{i,k}^{\rho b}(t)$, $j=1,\ldots,n_i,\ k=1,\ldots,n, k\neq i$. Similarly, $e_{i,2}^{r,act}$, $e_{i,2}^{\rho,act}$, and $e_{i,2}^{\rho b,act}$ give a measure for the actual stability margin relating to the magnitude of the respective variables. Thus, to observe the effect of the uncertainty bounds on the actual stability margins, $e_{i,l}^{\bullet,act}$s are calculated and depicted for a number of example cases.

Due to space limitations only one example case is included here. Further cases may be found in Munyas and İftar (2004, 2005a). Here, the network shown in Figure 4, which has three bottleneck nodes (N1, N2, and N3) with $n_1=2$, $n_2=3$, and $n_3=4$, is considered. The nominal time-delays and design parameters used are given in Tables 1 and 2, respectively (since $h_{i,j}^{rf}=h_{i,j}^{r}-h_{i,j}^{rb}$ and $h_{i,j}^{\rho f}=h_{i,j}^{\rho}-h_{i,j}^{\rho b}$, $h_{i,j}^{rf}$ and $h_{i,j}^{\rho f}$ are not shown in Table 1). In the calculation of the actual

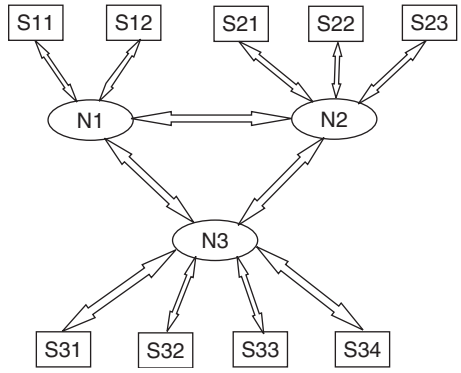

Figure 4. Example network.

Table 1. Nominal time-delays.

j	$h_{1,j}^{r}$	$h_{1,j}^{rb}$	$h_{1,j}^{\rho}$	$h_{1,j}^{\rho b}$	$h_{2,j}^{r}$	$h_{2,j}^{rb}$	$h_{2,j}^{\rho}$	$h_{2,j}^{\rho b}$	$h_{3,j}^{r}$	$h_{3,j}^{rb}$	$h_{3,j}^{\rho}$	$h_{3,j}^{\rho b}$
1	1.5	1	–	–	2.5	2	3	2	2.5	2	2	1
2	1.5	1	2	1	2.5	2	–	–	2.5	2	1	0.5
3	–	–	3	1.5	2.5	2	3.5	3	2.5	2	–	–
4	–	–	–	–	–	–	–	–	2.5	2	–	–

stability margins, only two parameters for each bottleneck node are changed because it is easy to visualise the effects of the bounds in 3D-plots. For the 1st bottleneck node, $\beta_{1,1}^{r}$ and $\delta_{1,1}^{r+}$ are changed from 0.001 to 0.999 and from 0.001 to 3.5, respectively; for the 2nd bottleneck node, $\beta_{1,2}^{\rho}$ and $\delta_{1,2}^{\rho+}$ are changed from 0.001 to 0.999 and from 0.001 to 4, respectively; and for the 3rd bottleneck node, $\beta_{3,2}^{\rho b}$ and $\delta_{3,2}^{\rho b+}$ are changed from 0.001 to 0.999 and from 0.001 to 3, respectively. Meanwhile, $\beta_{1,1}^{rf}=(1/2)\beta_{1,1}^{r}$, $\beta_{1,2}^{\rho f}=(1/2)\beta_{1,2}^{\rho}$ and all the other design parameters for the three bottleneck nodes are held constant at their design values given in Table 2. For cases in which $\beta_{i,j}^{\bullet f}$ is taken as equal to 0 or $\beta_{i,j}^{\bullet}$ and for cases where different network conditions and parameter values are considered, see Munyas and İftar (2004, 2005a).

The results are given in Figures 5–13. Figure 5 indicates that, as $\beta_{1,1}^{r}$, the design bound on $\dot{\delta}_{1,1}^{r}(t)$, is increased, the stability margin on $\dot{\delta}_{1,1}^{r}(t)$ increases, indicated by the increase in $e_{1,1}^{r,act}$. Figures 5–7 also indicate that, when $\beta_{1,1}^{r}$ is changed and all other uncertainty bounds are kept constant, the values of $e_{1,2}^{r,act}$, $e_{1,l}^{\rho,act}$ and $e_{1,l}^{\rho b,act}$, $l=1,2$, remain almost constant except when $\beta_{1,1}^{r}$ is made too close to 1. This indicates that the stability margins on $\delta_{1,j}^{r}(t)$, $\delta_{k,1}^{\rho}(t)$, $\delta_{1,k}^{\rho b}(t)$, $\dot{\delta}_{k,1}^{\rho}(t)$ and $\dot{\delta}_{1,k}^{\rho b}(t)$ $(j=1,\ldots,n_1,$ $k=1,\ldots,n, k\neq i)$ are insensitive to changes in $\beta_{1,1}^{r}$ except when $\beta_{1,1}^{r}$ is too close to 1. As $\beta_{1,1}^{r}$ gets close to 1, $\tilde{\gamma}_1$ increases without bounds, driving $e_{1,k}^{\bullet,act}$, except $e_{1,1}^{r,act}$, to zero. From Figures 5–7, we can

Table 2. Design parameters.

i,j	1, 1	1, 2	1, 3	2, 1	2, 2	2, 3	3, 1	3, 2	3, 3	3, 4
$\alpha^{r}_{i,j}$	0.1	0.15	–	0.2	0.15	0.05	0.08	0.12	0.06	0.09
$\alpha^{\rho}_{i,j}$	–	0.2	0.2	0.25	–	0.3	0.35	0.25	–	–
$\alpha^{\rho b}_{i,j}$	–	0.05	0.1	0.08	–	0.07	0.05	0.1	–	–
$\beta^{r}_{i,j}$	0.2	0.2	–	0.15	0.15	0.15	0.3	0.3	0.3	0.3
$\beta^{rf}_{i,j}$	0.02	0.02	–	0.03	0.03	0.03	0.04	0.04	0.04	0.04
$\beta^{\rho}_{i,j}$	–	0.25	0.3	0.3	–	0.33	0.4	0.1	–	–
$\beta^{\rho b}_{i,j}$	–	0.1	0.15	0.15	–	0.2	0.25	0.05	–	–
$\beta^{\rho f}_{i,j}$	–	0.15	0.15	0.15	–	0.13	0.15	0.05	–	–
$\delta^{r+}_{i,j}$	2	2	–	3	3	3	2.5	2.5	2.5	2.5
$\delta^{\rho+}_{i,j}$	–	2	3	3	–	3.5	2	1	–	–
$\delta^{\rho b+}_{i,j}$	–	1	1.5	2	–	3	1	0.5	–	–

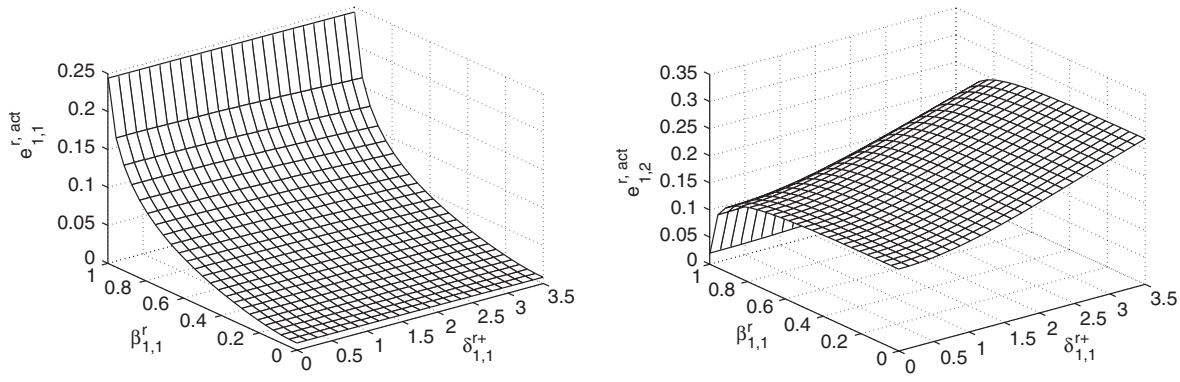

Figure 5. Stability margins $e^{r,act}_{1,1}$ and $e^{r,act}_{1,2}$.

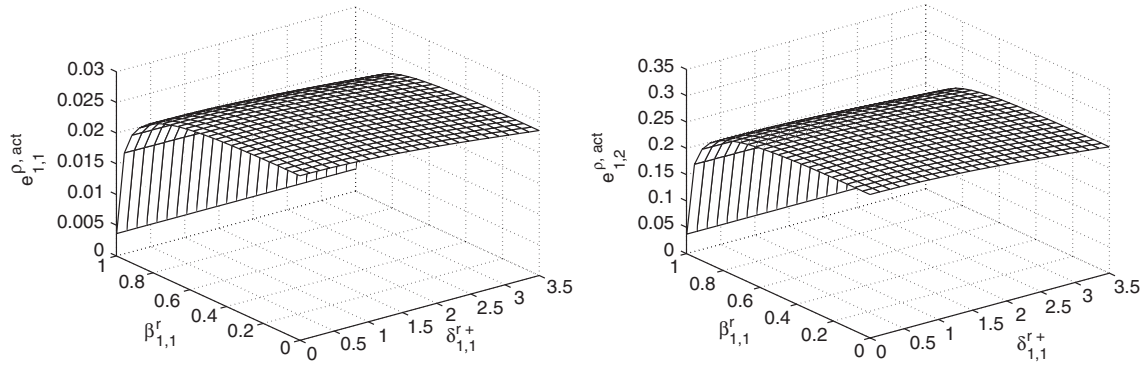

Figure 6. Stability margins $e^{\rho,act}_{1,1}$ and $e^{\rho,act}_{1,2}$.

further say that as $\delta^{r+}_{1,1}$, the design bound on $\delta^{r}_{1,1}(t)$, is increased, $e^{r,act}_{1,2}$ increases, but $e^{r,act}_{1,1}$, $e^{\rho,act}_{1,l}$ and $e^{\rho b,act}_{1,l}$, $l=1$, 2, remain almost constant as long as the other uncertainty bounds are kept constant. Similar conclusions are drawn from Figures 8–10 when r is replaced by ρ and from Figures 11–13 when r is replaced by ρb. The effects of changing the design bounds on the actual stability margins are summarised in Table 3, which is taken from Munyas and İftar (2005a). In this table, '+' means that the stability margin increases with increasing design bound, '–' means that the stability margin is insensitive to changes in the design bound, and '*' means that the stability margin is insensitive to changes in the design bound except when the bound gets too close to 1.

In conclusion, to have large stability margins, the uncertainty bounds $\delta^{\bullet+}_{i,j}$ and $\beta^{\bullet}_{i,j}$ should be chosen as large as possible ($\beta^{\bullet}_{i,j}$ should not be too close to 1). However, such a choice of the bounds lead to a smooth

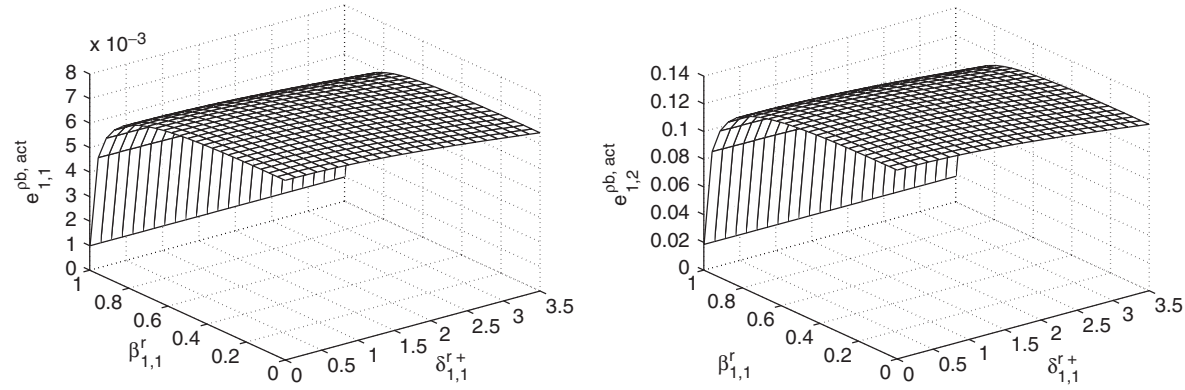

Figure 7. Stability margins $e_{1,1}^{\rho b,act}$ and $e_{1,2}^{\rho b,act}$.

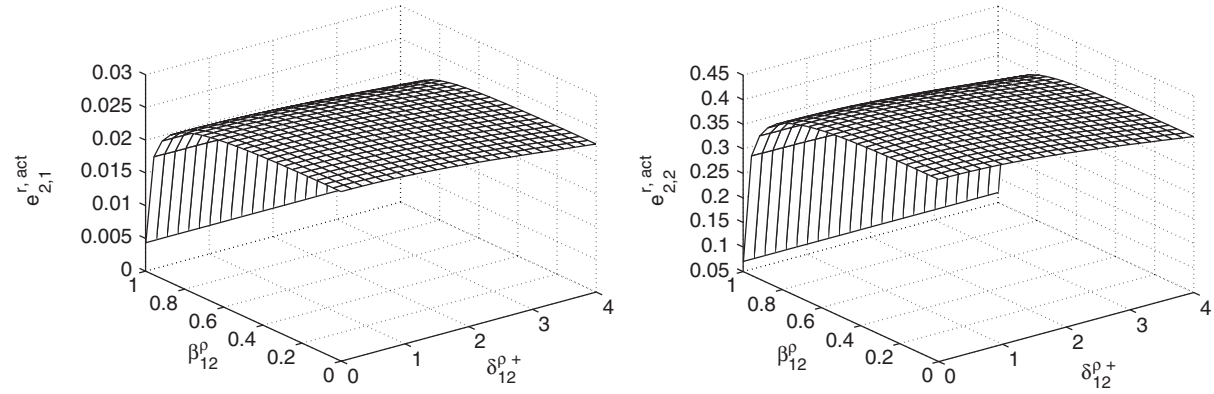

Figure 8. Stability margins $e_{2,1}^{r,act}$ and $e_{2,2}^{r,act}$.

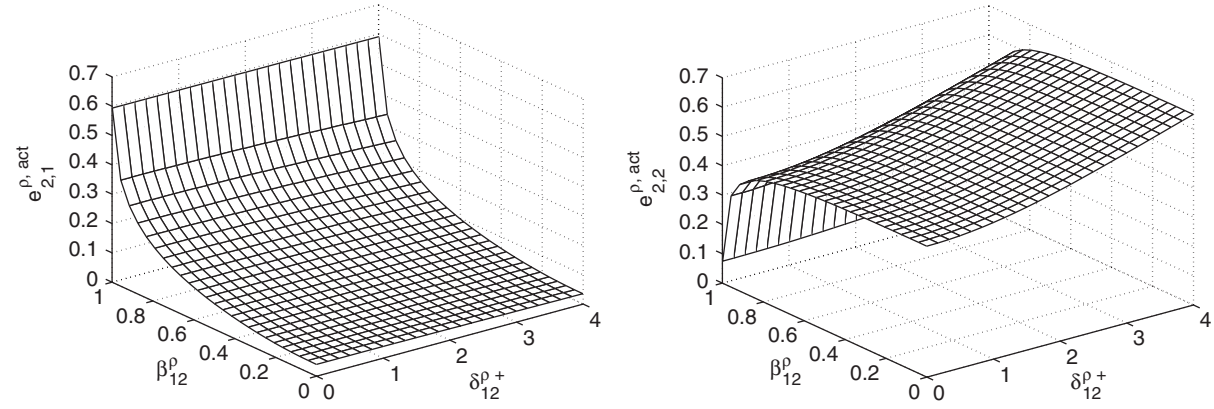

Figure 9. Stability margins $e_{2,1}^{\rho,act}$ and $e_{2,2}^{\rho,act}$.

but very slow response. When these bounds are chosen small, an oscillatory but faster response is obtained (see §6). Thus, here, there is a trade-off between robustness and the time-domain performance.

6. Simulation results

The network shown in Figure 4 under the decentralised controllers derived in §3 is implemented using MATLAB Simulink and its time domain performance is investigated under various conditions. Rather than using the fluid-flow network model used for controller design, however, we use a discrete model for all the simulations. We assume that data flow consists of discrete packets of size 1 Mbits each. All the links are assumed to have a physical capacity of 100 Mbits/second. Therefore, each data packet is modelled as a pulse of width

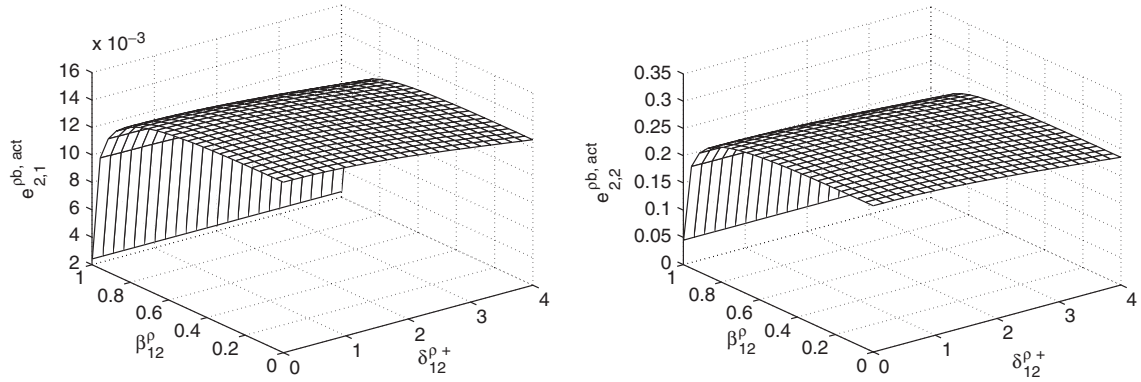

Figure 10. Stability margins $e^{\rho b,\,act}_{2,1}$ and $e^{\rho b,\,act}_{2,2}$.

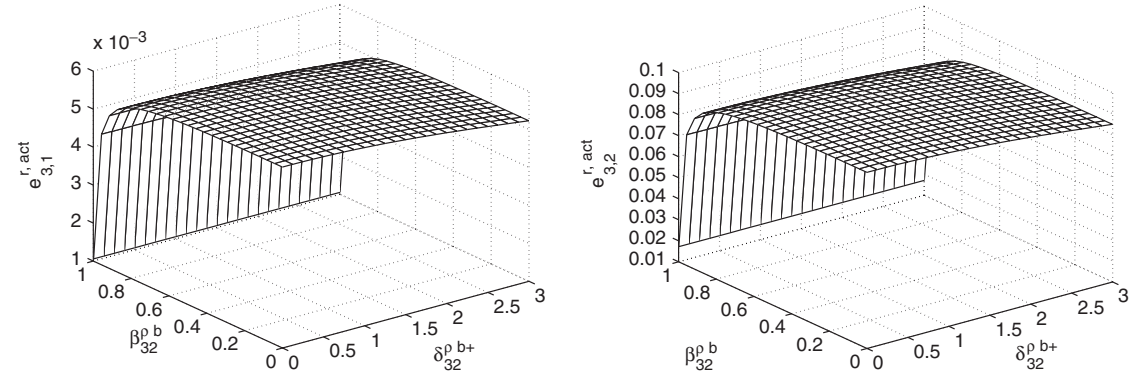

Figure 11. Stability margins $e^{r,\,act}_{3,1}$ and $e^{r,\,act}_{3,2}$.

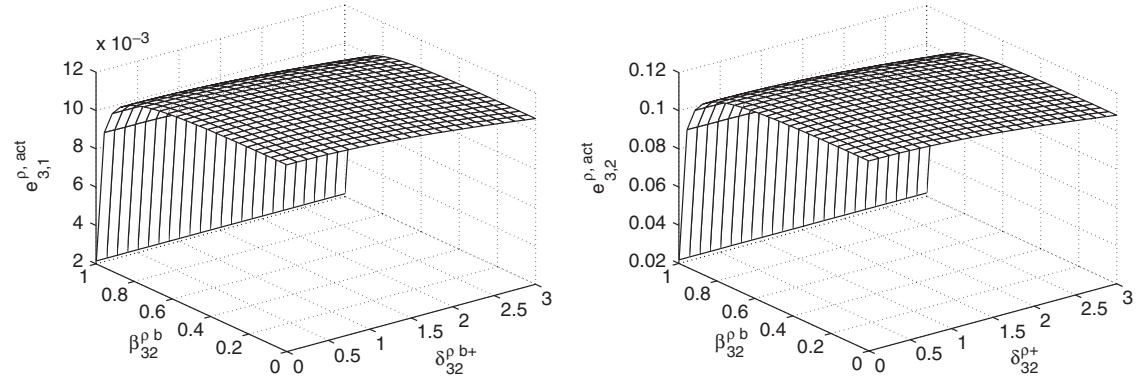

Figure 12. Stability margins $e^{\rho,\,act}_{3,1}$ and $e^{\rho,\,act}_{3,2}$.

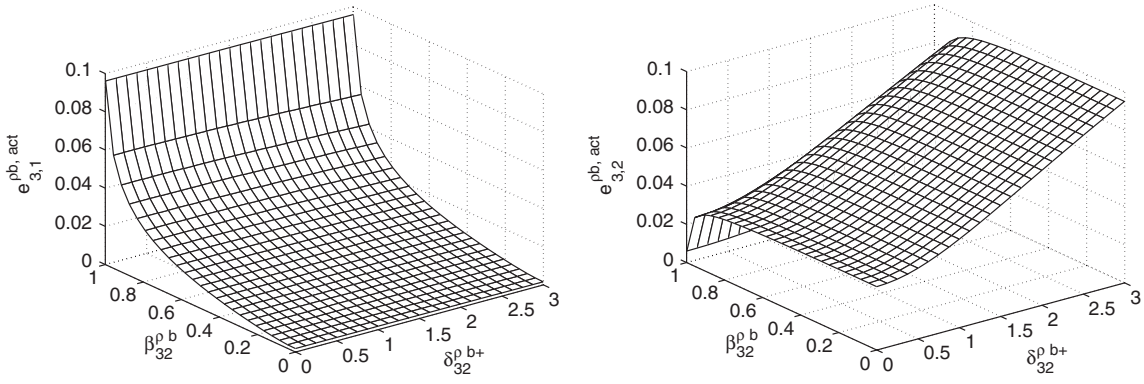

Figure 13. Stability margins $e^{\rho b,\,act}_{3,1}$ and $e^{\rho b,\,act}_{3,2}$.

10 milliseconds. Control packets, which carry rate information from each bottleneck node to each of its sources and the other bottlenecks, on the other hand, have much smaller sizes. The output of the controllers are assumed to be sampled at a rate of 0.1 kHz. That is, each bottleneck node sends a control packet to each of its sources and the other bottlenecks at every 10 milliseconds. Each source and each bottleneck node updates its data sending rate as soon as a new control packet arrives (if the current rate is r packets/second, then a packet of size 1 Mbits is sent every $(1/r)$ seconds). Note that, due to the presence of time-varying backward time-delays, control packets are not necessarily received and hence, data sending rates are not necessarily updated at equal intervals. A constant simulation step size of 2 milliseconds is used for all simulations.

Table 3. Effects of the design bounds on the stability margins.

	Stability margin on					
Design bound	$\delta^{r}_{i,j}(t)$	$\dot{\delta}^{r}_{i,j}(t)$	$\delta^{\rho}_{i,j}(t)$	$\dot{\delta}^{\rho}_{i,j}(t)$	$\delta^{\rho b}_{i,j}(t)$	$\dot{\delta}^{\rho b}_{i,j}(t)$
$\beta^{r}_{i,j}$	*	+	*	*	*	*
$\beta^{rf}_{i,j}$	*	+	*	*	*	*
$\beta^{\rho}_{i,j}$	*	*	*	+	*	*
$\beta^{\rho f}_{i,j}$	*	*	*	+	*	*
$\beta^{\rho b}_{i,j}$	*	*	*	*	*	+
$\delta^{r+}_{i,j}$	+	−	−	−	−	−
$\delta^{\rho+}_{i,j}$	−	−	+	−	−	−
$\delta^{\rho b+}_{i,j}$	−	−	−	−	+	−

The nominal time-delays (in seconds) given in Table 1 and design parameters given in Table 2 (except where indicated) are used in the simulations. Maximum outgoing flow rates, $c_i(t)$s, are the same and equal to 50 packets/second in all cases except in Case 5 (the actual outgoing flow rate at the bottleneck node i is equal to $c_i(t)$ at time t, if $q_i(t) > 0$; otherwise, i.e., if $q_i(t) = 0$, then any packet arriving to the buffer is immediately sent out at a rate not to exceed $c_i(t)$). Desired queue lengths, $q_{d,i}$s are the same and equal to 50 packets. The uncertain part of the actual time-delays are taken as $\delta^{rf}_{i,j}(t) = \delta^{\rho f}_{i,j}(t) = \delta^{rb}_{i,j}(t) = \delta^{\rho b}_{i,j}(t) = 0.05\sin((\pi/50)t)$ seconds (where t is also in seconds). The results are given in Figures 14–18. For all cases, the graphs (a), (b) and (c) show the queue lengths (in packets) and flow rates (in packets/second) versus time (in seconds) of the sources of the bottleneck nodes 1, 2 and 3, respectively, while graph (d) shows the flow rates versus time between the bottleneck nodes.

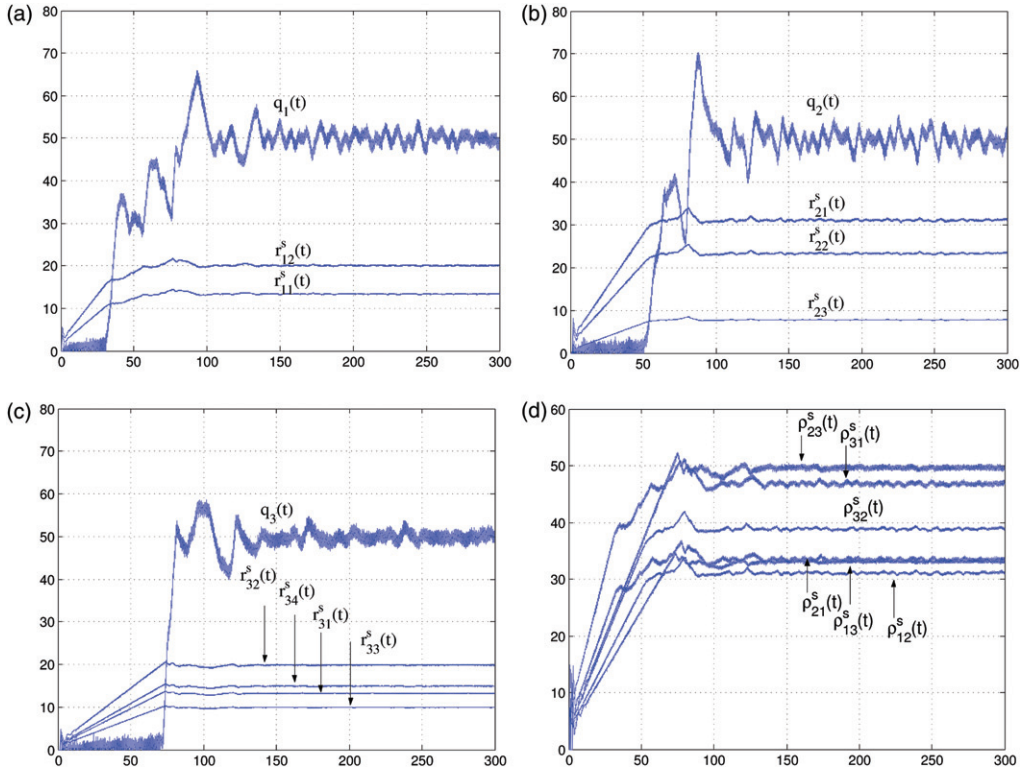

Figure 14. Results for Case 1.

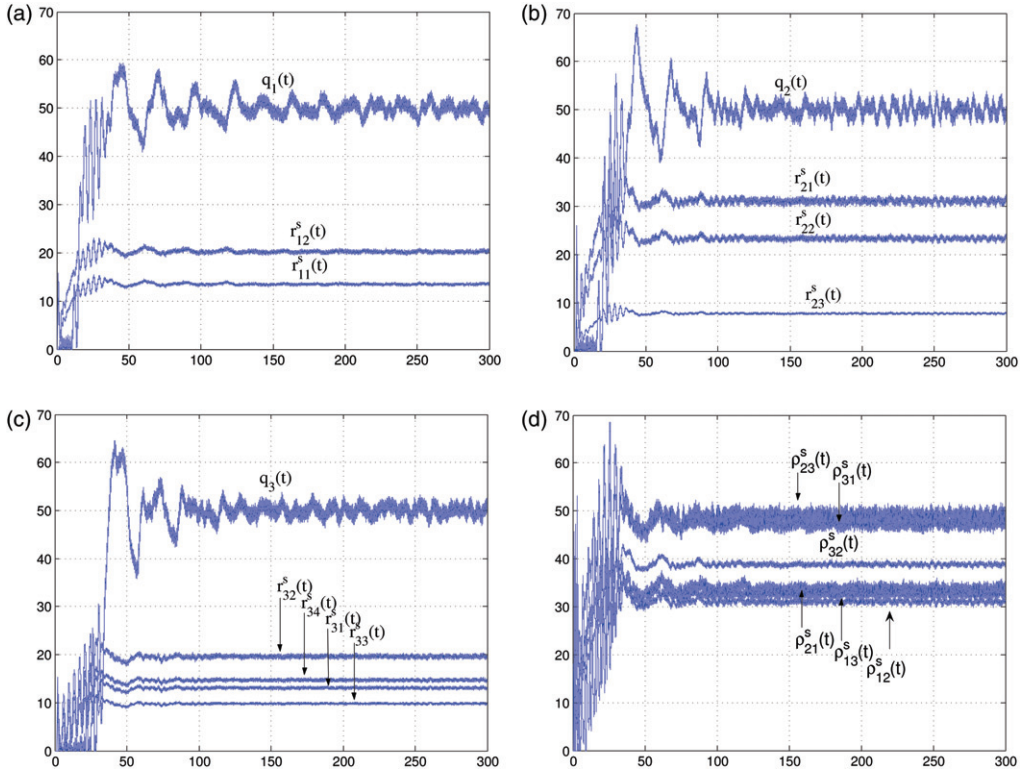

Figure 15. Results for Case 2.

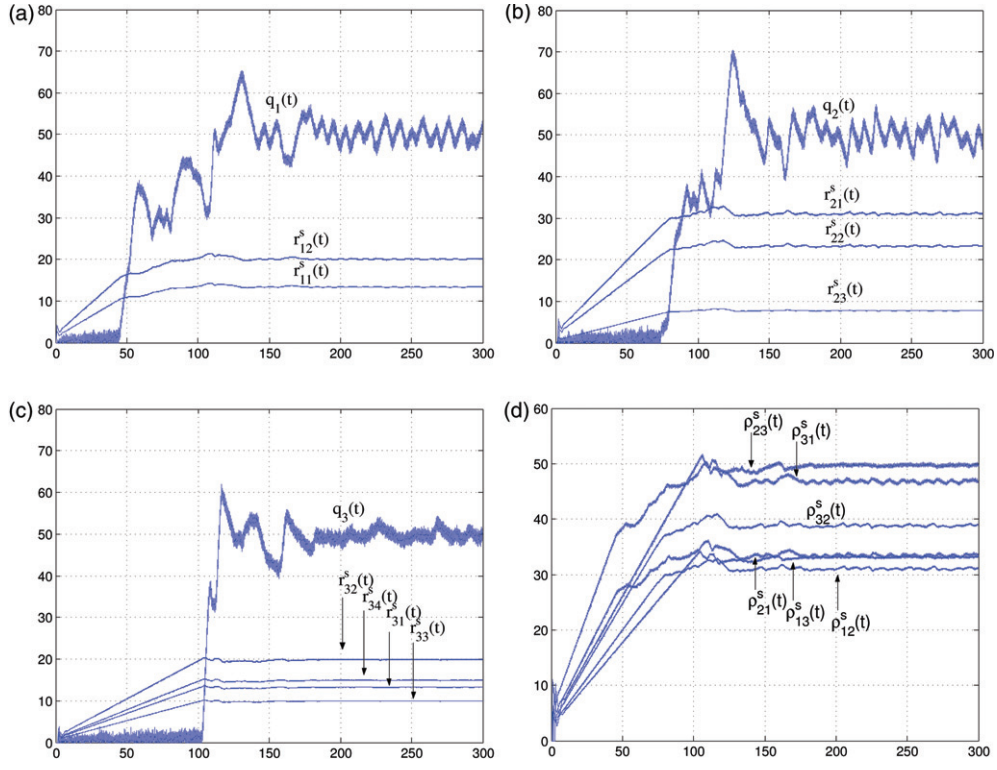

Figure 16. Results for Case 3.

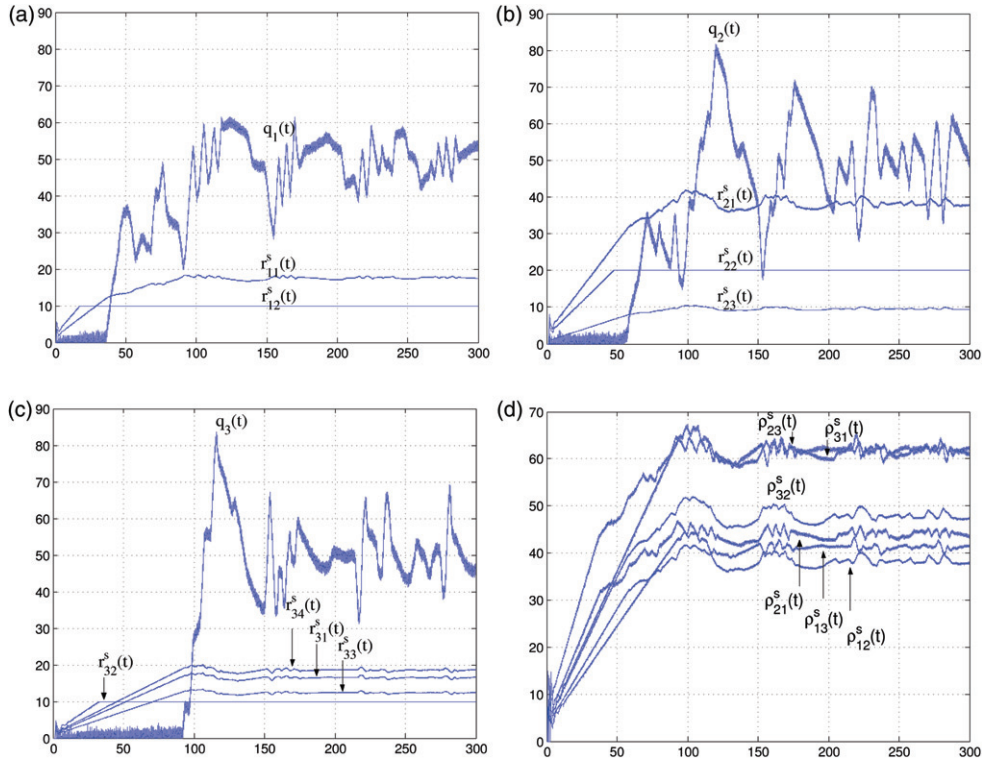

Figure 17. Results for Case 4.

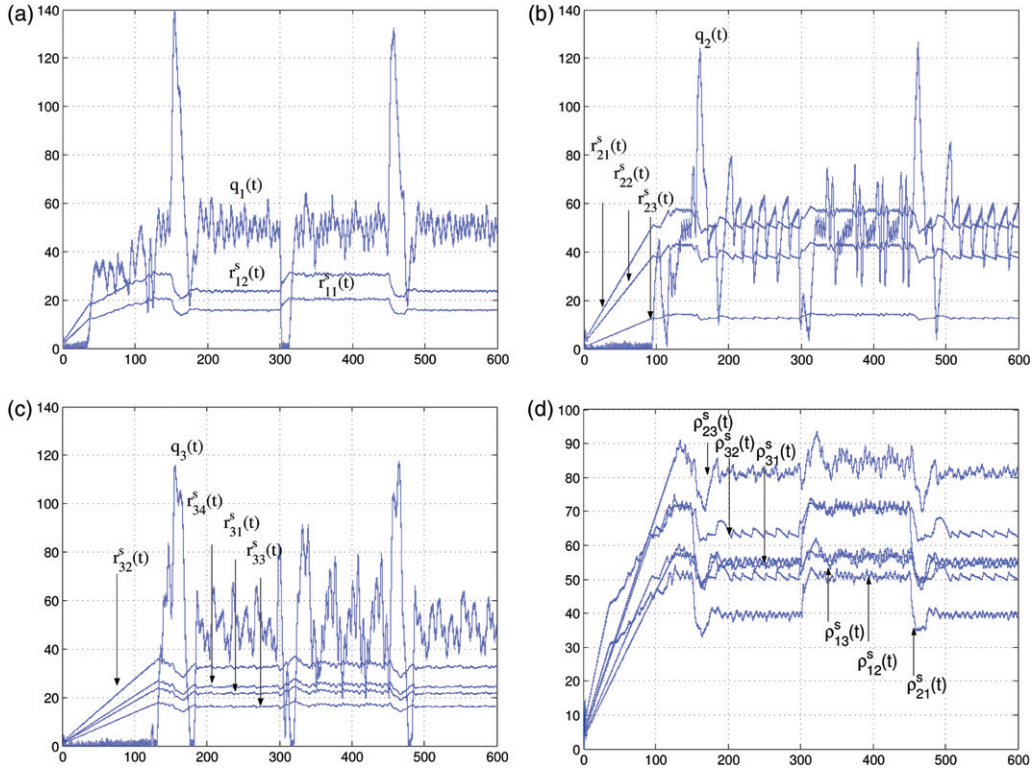

Figure 18. Results for Case 5.

Case 1: As shown in Figure 14, in all the bottleneck nodes, there is a duration where the queue length remains almost zero. This is the period needed for the sum of the incoming flows to exceed the capacity of the outgoing link at the bottleneck node. The high-frequency oscillations in the queue length are due to discrete arrival/departure of packets (those oscillations would not be seen if a fluid-flow model was used). Besides those oscillations, the existence of time-varying forward time-delays also causes oscillations, especially at the steady-state (the frequency of these oscillations is the same as the rate of change of the forward time-delays (0.01 Hz for this example)). All the queue lengths and flow rates reach an oscillatory steady-state within 100–150 seconds. We note that the time-average value of the steady-state flow rates satisfies fairness conditions (24)–(25) and (44) in this and all other cases, except in Case 4, below. The time-average value of the queue lengths at the steady-state are always equal to the desired queue length, $q_{d,i}$.

Case 2: In this case the values of $\delta^{r+}_{i,j}$, $\delta^{\rho+}_{i,j}$, $\delta^{\rho b+}_{i,j}$, $\beta^{r}_{i,j}$, $\beta^{rf}_{i,j}$, $\beta^{\rho}_{i,j}$, $\beta^{\rho b}_{i,j}$, $\beta^{\rho f}_{i,j}$ are decreased to one eighth of the values given in Table 2. As shown in Figure 15, this causes an oscillatory (near unstable) transient response. The response in this case, however, is much faster than the response obtained in Case 1.

Case 3: $\delta^{r+}_{i,j}$, $\delta^{\rho+}_{i,j}$, $\delta^{\rho b+}_{i,j}$, $\beta^{r}_{i,j}$, $\beta^{rf}_{i,j}$, $\beta^{\rho}_{i,j}$, $\beta^{\rho b}_{i,j}$, $\beta^{\rho f}_{i,j}$ values are twice the ones given in Table 2. As shown in Figure 16, this causes a slower but smoother transient response compared to cases 1 and 2. The magnitude of the steady-state oscillations are also larger due to the slowness of the response.

Case 4: Data supplying rates of the sources are limited by some $d_{i,j}$ values as shown in Table 4. As shown in Figure 17, the values of $r^{s}_{1,2}(t)$, $r^{s}_{2,2}(t)$ and $r^{s}_{3,2}(t)$ are saturated by these limits. The controllers, however, can successfully redistribute the unused rates to other sources and other bottlenecks. The system reaches a desired steady-state, although the transient response is more oscillatory and takes a longer time. The fairness conditions (24)–(25) and (44) are naturally not satisfied in this case since some of the rates are saturated. The time-average steady-state value of the unsaturated rates, however, are distributed according to their fairness weights among themselves.

Table 4. Rate limits for Case 4.

i,j	1, 1	1, 2	2, 1	2, 2	2, 3	3, 1	3, 2	3, 3	3, 4
$d_{i,j}$ (packets/second)	50	10	50	20	50	50	10	50	50

Case 5: The maximum outgoing flow rate at the first bottleneck node, $c_1(t)$, switches between 60 packets/second and 30 packets/second as a square wave of period 300 seconds. The outgoing flow rates at the other bottleneck nodes, $c_2(t)$ and $c_3(t)$, are constant and equal to 100 packets/second. As a result of these changes in $c_1(t)$, all the queue lengths and flow rates go through transients in every change of $c_1(t)$ as shown in Figure 18 (note the difference in the time scale of this figure compared to other figures). The system, however, reaches the desired steady-state before the next change in $c_1(t)$.

7. Conclusion

In this work, we have considered decentralised rate-based flow controller design in multi-bottleneck networks. The considered multi-input multi-output $\mathcal{H}^{\infty}$ optimisation problem was set forth in Biberović et al. (2001). We solved a suboptimal version of this problem using a series of single-input single-output $\mathcal{H}^{\infty}$ optimisations for which the method of Toker and Özbay (1995) is applied. We presented the implementation of the decentralised controllers at different bottleneck nodes. The controllers are robust to time-varying uncertain multiple time-delays in different channels and also satisfy tracking and weighted fairness requirements.

The stability margins for uncertainties in the multiple time-delays and for the rate of change of the time-delays have also been considered and the lower bounds for the stability margins have been derived. According to the sufficient conditions obtained the lower bounds on the actual stability margins were depicted with respect to the design bounds on the uncertainties for various cases. The results show that when the design bounds on the magnitude of the uncertainties and the rate of change of the uncertainties in the time-delays are increased, the corresponding stability margins also increase and hence, the system is highly robust to time-varying time-delays. However, the results of the simulations illustrating the time-domain performance of the proposed controllers indicate that the controller designed with large values of the design bounds will be conservative, resulting in a slow queue response. To get a faster response, the design bounds can be chosen small, but in this case, the transient response becomes more oscillatory.

Although we used a fluid-flow model and ignored all the hard constraints for the purpose of controller design, the simulations in §6 indicate that the controller works well when applied to a network where the data flow is discrete and hard contraints are present. The performance evaluation of the proposed

controller when applied to a real network could be the subject of a future work.

The control approach presented in this work assumes that each bottleneck node acts as a virtual source for the next bottleneck node on the path of a connection. In this approach, there is a control loop between each successive bottleneck node as well as between the actual source and the first bottleneck node. As opposed to the case where the control loop is between a bottleneck and the actual source, the present approach allows better use of network resources, by reducing the length of a control loop.

The flow dynamics represented by (1) may also appear in other commodity flow problems, such as transportation networks, material transport systems (e.g. oil or gas pipelines), process control, and manufacturing systems. Of course, there may be differences between such systems and data-communication networks considered here, when it comes to detailed modelling. However, whenever the flow dynamics of such systems can be modelled by simple time-delays (possibly time-varying and uncertain) and commodities can be stored freely at certain places, the approach presented here may directly be applied (see Remark 2). Furthermore, even if a more detailed model (e.g., a model derived by using Navier-Stokes equations for a gas transport system) is used for a system which involve time-delays, the ideas presented in this work could be a good starting point to attack the flow controller design problem. Moreover, note that, model (1) is simply an interconnection of multivariable integrating systems with uncertain and time-varying time-delays. Therefore, the presented approach may find application, not only in flow control problems, but in control of other interconnected multivariable integrating systems with uncertain and time-varying time-delays.

Acknowledgements

Programs used in the simulations are based on the programs previously developed by Taesam Kang, Pierre-Francois Quet, and Banu Ataşlar.

References

Altman, E., Başar, T., and Srikant, R. (1997), 'Multi-User Rate-based Flow Control with Action Delays: A Team-theoretic Approach', in *Proceedings of ACM/SIGCOMM*, San Diego, CA, U.S.A., pp. 2387–2392.

BenMohamed, L., and Meerkov, S. (1993), 'Feedback Control of Congestion in Store-and-forward Datagram Networks: The Case of a Single Congested Node', *IEEE/ACM Transactions on Networking*, 1, 693–708.

BenMohamed, L., and Meerkov, S. (1997), 'Feedback Control of Congestion in Packet Switching Networks: The Case of Multiple Congested Nodes', *International Journal on Communication Systems*, 10, 227–246.

Biberović, E. (2001), 'Flow Control in High-Speed Data Communication Networks', Master's thesis, Anadolu University, Eskişehir, Turkey (in Turkish).

Biberović, E., İftar, A., and Özbay, H. (2001) 'A Solution to the Roubust Flow Control Problem for Networks with Multiple Bottlenecks', in *Proceedings of the 40th IEEE Conference on Decision and Control*, Orlando, FL, U.S.A., pp. 2303–2308.

Bonomi, F., and Fendick, K.W. (1995), 'The Rate-based Flow Control Framework for the Available Bit Rate ATM Service', *IEEE Network Magazine*, 25, 24–39.

Cavendish, D., Gerla, M., and Mascolo, S. (2004), 'A Control Theoretical Approach to Congestion Control in Packet Networks', *IEEE Transactions on Networking*, 12, 893–906.

Floyd, S. (1994), 'End-to-end Congestion Control Schemes: Utility Functions, Random Losses and ECN Marks', *ACM Computer Communication Review*, 24, 10–23.

Floyd, S., Handley, M., Padhye, J., and Widmer, J. (2000), 'Equation-based Congestion Control for Unicast Applications', in *Proceedings of ACM/SIGCOMM*, Stockholm, Sweden, pp. 43–46.

Kung, H.T., and Morris, R. (1995), 'Credit Based Flow Control for ATM Networks', *IEEE Network Magazine*, 9, 40–48.

Kunniyur, S., and Srikant, R. (2000), 'TCP and Explicit Congestion Notification', in *Proceedings of the INFOCOM 2000*, Tel Aviv, Israel, pp. 1323–1332.

Laberteaux, K.P., Rohrs, C.E., and Antsaklis, P.J. (2002), 'A Practical Controller for Explicit Rate Congestion Control', *IEEE Transactions on Automatic Control*, 47, 960–978.

Mascolo, S. (2000), 'Smith's Principle for Congestion Control in High-speed Data Networks', *IEEE Transactions on Automatic Control*, 45, 358–364.

Mascolo, S., and Cavendish, D. (1996), 'ATM Rate Based Congestion Control Using a Smith Predictor: An EPRCA Implementation', in *Proceedings of the GLOBECOM'96*, London, U.K, pp. 569–576.

Munyas, İ., and İftar, A. (2004), 'Stability Margins for a Rate-based Flow Control Problem in Multiple Bottleneck Networks', Technical Report, Department of Electrical and Electronics Engineering, Anadolu University, Eskişehir, Turkey.

Munyas, İ., and İftar, A. (2005a) 'Stability Margins for a Rate-based Flow Control Problem in Multiple Bottleneck Network', in *Proceedings of the IFAC World Congress*, Prague, Czech Republic.

Munyas, İ., and İftar, A. (2005b) '$\mathcal{H}^{\infty}$-based Flow Control for ATM Networks with Multiple Bottlenecks', in *Proceedings of the IFAC World Congress*, Prague, Czech Republic.

Munyas, İ., Yelbaşi, Ö., and İftar, A. (2003) 'Decentralized Robust Flow Controller Design for Networks with Multiple Bottlenecks', in *Proceedings of the the European Control Conference*, Cambridge, U.K.

Ohsaki, H., Murata, M., Suzuki, H., Ikeda, C., and Miyahara, H. (1995a), 'Analysis of Rate-based Congestion Control Algorithms for ATM Networks-part I: Steady State Analysis', in *Proceedings of the GLOBECOM'95*, Singapore, pp. 296–303.

Ohsaki, H., Murata, M., Suzuki, H., Ikeda, C., and Miyahara, H. (1995b), 'Analysis of Rate-based Congestion Control Algorithms for ATM Networks -Part II: Inital Transient State Analysis', in *Proceedings of the GLOBECOM'95*, Singapore, pp. 1089–1094.

Quet, P.-F., Ataşlar, B., İftar, A., Özbay, H., Kalyanaraman, S., and Kang, T. (2002), 'Rate-based Flow Controllers for Communication Networks in the Presence of Uncertain Time-varying Multiple Time Delays', *Automatica*, 38, 917–928.

Srikant, R. (2004), *The Mathematics of Internet Congestion Control*, Boston, MA: Birkhäuser.

Toker, O., and Özbay, H. (1995), '$\mathcal{H}^\infty$ Optimal and Suboptimal Controllers for Infinite Dimensional SISO Plants', *IEEE Transactions on Automatic Control*, 40, 751–755.

Zhou, K., Doyle, J.C., and Glover, K. (1995), *Robust and Optimal Control*, Upper Saddle River, NJ: Prentice Hall Inc.

Appendices

A. *Control problem setup*

By integrating (1) and substituting the expressions of data receiving rates (4) and (5), the queue length at the ith bottleneck node becomes,

$$q_i(t)=\sum_{j=1}^{n_i}\left[\int_0^t\left(1-\dot{\delta}_{i,j}^{rf}(\nu)\right)r_{i,j}\left(\nu-\tau_{i,j}(\nu)\right)\mathrm{d}\nu\right]-\int_0^t c_i(\nu)\mathrm{d}\nu+q_i(0)$$
$$+\sum_{j=1,\,j\neq i}^{n}\int_0^t\left[\left(1-\dot{\delta}_{j,i}^{\rho f}(\nu)\right)\rho_{j,i}\left(\nu-\phi_{j,i}(\nu)\right)-\rho_{i,j}\left(\nu-\phi_{i,j}^{b}(\nu)\right)\right]\mathrm{d}\nu. \tag{29}$$

The nominal queue length is then obtained by setting all uncertainties to zero

$$q_{o,i}(t)=\sum_{j=1}^{n_i}\int_0^t r_{i,j}\left(\nu-h_{i,j}^{r}\right)\mathrm{d}\nu-\int_0^t c_i(\nu)\mathrm{d}\nu+q_i(0)$$
$$+\sum_{j=1,\,j\neq i}^{n}\int_0^t\left[\rho_{j,i}\left(\nu-h_{j,i}^{\rho}\right)-\rho_{i,j}\left(\nu-h_{i,j}^{\rho b}\right)\right]\mathrm{d}\nu. \tag{30}$$

By setting the initial condition $q_i(0)$ to zero, taking the Laplace transform of both sides of (30), separating the term $-(1/s)c_i(s)$, as it is taken care of separately in Figure 2, and defining the output of the controller K as

$$u:=\begin{bmatrix}r\\ \rho\end{bmatrix}:=\begin{bmatrix}r_1\\ \vdots\\ r_n\\ \rho_1\\ \vdots\\ \rho_n\end{bmatrix},\qquad r_i:=\begin{bmatrix}r_{i,1}\\ \vdots\\ r_{i,n_i}\end{bmatrix},\qquad \rho_i:=\begin{bmatrix}\rho_{1,i}\\ \vdots\\ \rho_{(i-1),i}\\ \rho_{(i+1),i}\\ \vdots\\ \rho_{n,i}\end{bmatrix}, \tag{31}$$

the expression for the nominal plant is obtained as $P_o(s)=(1/s)\bar{P}e^{-Hs}\hat{P}$. Here

$$\hat{P}:=\begin{bmatrix}I_m & 0\\ 0 & \frac{1}{\sqrt{2}}I_{n(n-1)}\\ 0 & -\frac{1}{\sqrt{2}}I_{n(n-1)}\end{bmatrix},$$

where I_k denotes the $k\times k$ identity matrix and $m:=\sum_n^{i=1}n_i$. Furthermore, $H:=\mathrm{diag}(\bar{h})$, where

$$\bar{h}:=\begin{bmatrix}h^r & h^\rho & h^{\rho b}\end{bmatrix}$$
$$:=\begin{bmatrix}h_1^r & \cdots & h_n^r & h_1^\rho & \cdots & h_n^\rho & h_1^{\rho b} & \cdots & h_n^{\rho b}\end{bmatrix}$$

where $h_i^r:=[\,h_{i,1}^r\ \cdots\ h_{i,n_i}^r\,]$, $h_i^\rho:=[\,h_{1,i}^\rho\ \cdots\ h_{i-1,i}^\rho\ \ h_{i+1,i}^\rho\ \cdots\ h_{n,i}^\rho\,]$, and $h_i^{\rho b}:=[\,h_{1,i}^{\rho b}\ \cdots\ h_{i-1,i}^{\rho b}\ \ h_{i+1,i}^{\rho b}\ \cdots\ h_{n,i}^{\rho b}\,]$. Moreover, $\bar{P}:=[\,\bar{P}_r\ \ \sqrt{2}\bar{P}_\rho\ \ \sqrt{2}\bar{P}_{\rho b}\,]$, where $\bar{P}_r:=\mathrm{blockdiag}(\mathbf{1}_{n_1},\ldots,\mathbf{1}_{n_n})$, where $\mathbf{1}_k$ denotes the $1\times k$ dimensional row vector of 1s, $\bar{P}_\rho:=I_n\otimes\mathbf{1}_{n-1}$, where $\otimes$ denotes the Kronecker product, and

$$\bar{P}_{\rho b}:=\begin{bmatrix}0 & J_1 & J_1 & J_1 & \cdots & J_1 & J_1\\ J_1 & 0 & J_2 & J_2 & \cdots & J_2 & J_2\\ J_2 & J_2 & 0 & J_3 & \cdots & J_3 & J_3\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ J_{n-1} & J_{n-1} & J_{n-1} & J_{n-1} & \cdots & J_{n-1} & 0\end{bmatrix},$$

$$J_i:=\begin{bmatrix}0_{1\times(i-1)} & 1 & 0_{1\times(n-i-1)}\end{bmatrix}.$$

Next, to obtain the uncertainty model, the uncertainty in the queue length can be obtained as $\delta_{q_i}(t):=q_i(t)-q_{o,i}(t)$. Using (29) and (30) and following some manipulations (see Biberović (2001), for details), we obtain,

$$\delta_{q_i}(t)=\sum_{j=1}^{n_i}\bar{\delta}_{q_i}^{j}(t)+\sum_{j=1,\,j\neq i}^{n}\hat{\delta}_{q_i}^{j}(t)+\sum_{j=1,\,j\neq i}^{n}\tilde{\delta}_{q_i}^{j}(t) \tag{32}$$

where

$$\bar{\delta}_{q_i}^{j}(t)=\int_0^t\left(\dot{\delta}_{i,j}^{r}\left(\nu-\tau_{i,j}(\nu)\right)-\dot{\delta}_{i,j}^{rf}(\nu)\right)r_{i,j}\left(\nu-\tau_{i,j}(\nu)\right)\mathrm{d}\nu$$
$$-\int_{t-h_{i,j}^{r}-\delta_{i,j}^{r}(t)}^{t-h_{i,j}^{r}}r_{i,j}(\nu)\mathrm{d}\nu,$$

$$\hat{\delta}_{q_i}^{j}(t)=\int_0^t\left(\dot{\delta}_{j,i}^{\rho}\left(\nu-\phi_{j,i}(\nu)\right)-\dot{\delta}_{j,i}^{\rho f}(\nu)\right)\rho_{j,i}\left(\nu-\phi_{j,i}(\nu)\right)\mathrm{d}\nu$$
$$-\int_{t-h_{j,i}^{\rho}-\delta_{j,i}^{\rho}(t)}^{t-h_{j,i}^{\rho}}\rho_{j,i}(\nu)\mathrm{d}\nu,$$

$$\tilde{\delta}_{q_i}^{j}(t)=-\int_0^t\dot{\delta}_{i,j}^{\rho b}\left(\nu-\phi_{i,j}^{b}(\nu)\right)\rho_{i,j}\left(\nu-\phi_{i,j}^{b}(\nu)\right)\mathrm{d}\nu$$
$$+\int_{t-h_{i,j}^{\rho b}-\delta_{i,j}^{\rho b}(t)}^{t-h_{i,j}^{\rho b}}\rho_{i,j}(\nu)\mathrm{d}\nu.$$

The uncertain parts $\bar{\delta}_{q_i}^{j}(t)$, $\hat{\delta}_{q_i}^{j}(t)$ and $\tilde{\delta}_{q_i}^{j}(t)$ are generated by the systems shown in Figure 19, Figure 20, and Figure 21, respectively. In these figures, $M_\diamond$ represents multiplication by $\diamond$.

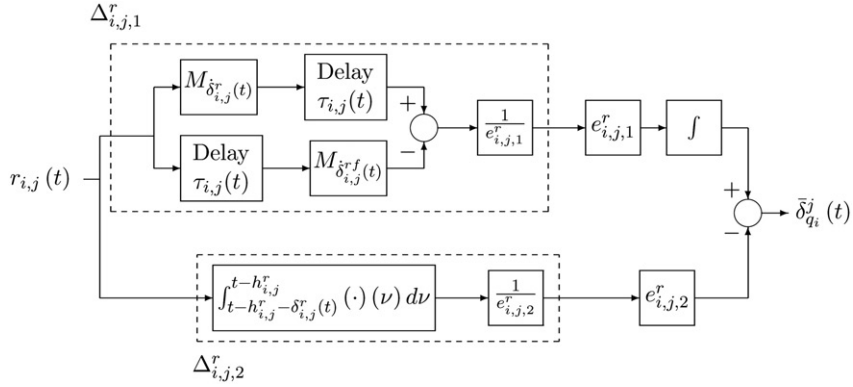

Figure 19. Uncertainties in the system, $\bar{\delta}^j_{q_i}(t)$.

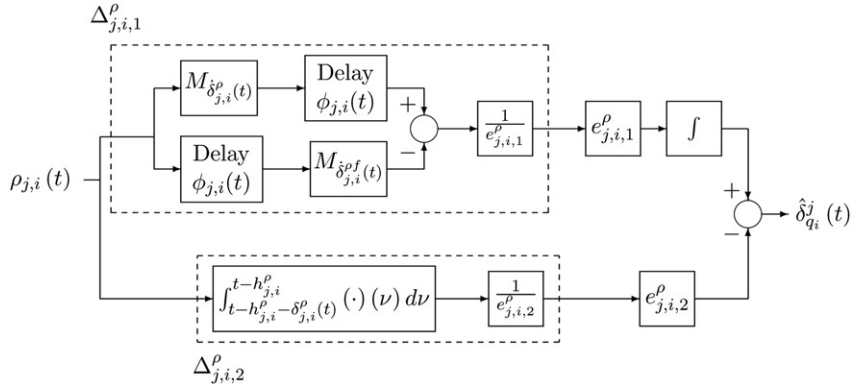

Figure 20. Uncertainties in the system, $\hat{\delta}^j_{q_i}(t)$.

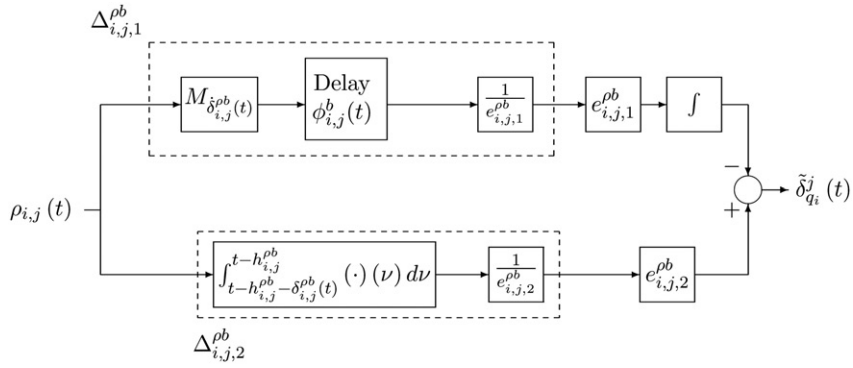

Figure 21. Uncertainties in the system, $\tilde{\delta}^j_{q_i}(t)$.

As in Quet et al. (2002), it can be shown that the $\mathcal{L}_2$-induced norm of the delay blocks in $\Delta^r_{i,j,1}$, $\Delta^\rho_{j,i,1}$ and $\Delta^{\rho b}_{i,j,1}$, in Figure 19, Figure 20, and Figure 21, respectively, is less than $(1/\sqrt{1-\beta^r_{i,j}})$, $(1/\sqrt{1-\beta^\rho_{j,i}})$, and $(1/\sqrt{1-\beta^{\rho b}_{i,j}})$, respectively. Thus, noting that the norms of the multiplication blocks are bounded by the bounds of the multipliers, the $\mathcal{L}_2$-induced norm of $\Delta^r_{i,j,1}$, $\Delta^\rho_{j,i,1}$, and $\Delta^{\rho b}_{i,j,1}$ become less than $((\beta^r_{i,j}+\beta^{rf}_{i,j})/\sqrt{1-\beta^r_{i,j}})(1/e^r_{i,j,1})$, $((\beta^\rho_{j,i}+\beta^{\rho f}_{j,i})/\sqrt{1-\beta^\rho_{j,i}})(1/e^\rho_{j,i,1})$, and $(\beta^{\rho b}_{i,j}/\sqrt{1-\beta^{\rho b}_{i,j}})(1/e^{\rho b}_{i,j,1})$, respectively. Thus, if $e^r_{i,j,1}=(\sqrt{2}(\beta^r_{i,j}+\beta^{rf}_{i,j})/\sqrt{1-\beta^r_{i,j}})$, $e^\rho_{j,i,1}=(\sqrt{2}(\beta^\rho_{j,i}+\beta^{\rho f}_{j,i})/\sqrt{1-\beta^\rho_{j,i}})$, and $e^{\rho b}_{i,j,1}=(\sqrt{2}\beta^{\rho b}_{i,j}/\sqrt{1-\beta^{\rho b}_{i,j}})$ are chosen, then the $\mathcal{L}_2$-induced norm of each of $\Delta^r_{i,j,1}$, $\Delta^\rho_{j,i,1}$, and $\Delta^{\rho b}_{i,j,1}$ is less than $(1/\sqrt{2})$.

The $\mathcal{L}_2$-induced norm of the blocks $\Delta^r_{i,j,2}$, $\Delta^\rho_{j,i,2}$, and $\Delta^{\rho b}_{i,j,2}$, in Figure 19, Figure 20, and Figure 21, respectively, can be calculated to be less than $(2\delta^{r+}_{i,j}/e^r_{i,j,2})$, $(2\delta^{\rho+}_{j,i}/e^\rho_{j,i,2})$, and

$(2\delta^{\rho b+}_{i,j}/e^{\rho b}_{i,j,2})$, respectively, as in Quet et al. (2002). Thus, choosing $e^{r}_{i,j,2} = 2\sqrt{2}\delta^{r+}_{i,j}$, $e^{\rho}_{j,i,2} = 2\sqrt{2}\delta^{\rho+}_{j,i}$, and $e^{\rho b}_{i,j,2} = 2\sqrt{2}\delta^{\rho b+}_{i,j}$, the $\mathcal{L}_2$-induced norm of each of these blocks becomes less than $(1/\sqrt{2})$, as well.

By combining Figure 19, Figure 20, and Figure 21, the operator from u to $\delta_q := [\,\delta_{q_1} \;\cdots\; \delta_{q_n}\,]^T$ can be written as $W_{21}\Delta^o_{LTV}W_{22}$, where

$$W_{22} := \begin{bmatrix} I_m & 0 \\ 0 & \frac{1}{\sqrt{2}} I_{n(n-1)} \\ 0 & \frac{1}{\sqrt{2}} I_{n(n-1)} \end{bmatrix},$$

$\Delta^o_{LTV} := \text{blockdiag}(\Delta^r, \Delta^\rho, \Delta^{\rho b})$, where $\Delta^r := \text{blockdiag}(\Delta^r_1, \ldots, \Delta^r_n)$,

$$\Delta^r_i := \text{blockdiag}\left(\begin{bmatrix} \Delta^r_{i,1,1} \\ \Delta^r_{i,1,2} \end{bmatrix}, \ldots, \begin{bmatrix} \Delta^r_{i,n_i,1} \\ \Delta^r_{i,n_i,2} \end{bmatrix}\right),$$

$$\Delta^\bullet := \text{blockdiag}\left(\Delta^\bullet_1, \ldots, \Delta^\bullet_n\right),$$

and

$$\Delta^\bullet_i := \text{blockdiag}\left(\begin{bmatrix} \Delta^\bullet_{1,i,1} \\ \Delta^\bullet_{1,i,2} \end{bmatrix}, \ldots, \begin{bmatrix} \Delta^\bullet_{i-1,i,1} \\ \Delta^\bullet_{i-1,i,2} \end{bmatrix}, \begin{bmatrix} \Delta^\bullet_{i+1,i,1} \\ \Delta^\bullet_{i+1,i,2} \end{bmatrix}, \ldots, \begin{bmatrix} \Delta^\bullet_{n,i,1} \\ \Delta^\bullet_{n,i,2} \end{bmatrix}\right),$$

where the superscript $\bullet$ represents ρ or ρb. Furthermore, $W_{21} := L \cdot \text{diag}(U)$, where $L := [\,L_r \;\; L_\rho \;\; L_{\rho b}\,]$ and $U := [\,U^r \;\; \sqrt{2}U^\rho \;\; \sqrt{2}U^{\rho b}\,]$, where $L_r := \text{blockdiag}(\mathbf{1}_{2n_1}, \ldots, \mathbf{1}_{2n_n})$, $L_\rho := \mathbf{1}_{2(n-1)} \otimes I_n$,

$$L_{\rho b} := \begin{bmatrix} 0 & \hat{J}_1 & \hat{J}_1 & \hat{J}_1 & \cdots & \hat{J}_1 & \hat{J}_1 \\ \hat{J}_1 & 0 & \hat{J}_2 & \hat{J}_2 & \cdots & \hat{J}_2 & \hat{J}_2 \\ \hat{J}_2 & \hat{J}_2 & 0 & \hat{J}_3 & \cdots & \hat{J}_3 & \hat{J}_3 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \hat{J}_{n-1} & \hat{J}_{n-1} & \hat{J}_{n-1} & \hat{J}_{n-1} & \cdots & \hat{J}_{n-1} & 0 \end{bmatrix},$$

where $\hat{J}_i := J_i \otimes \mathbf{1}_2$,

$$U^r := \begin{bmatrix} U^r_1 & U^r_2 & \cdots & U^r_n \end{bmatrix}, \quad U^r_i := \begin{bmatrix} U^r_{i,1} & U^r_{i,2} & \cdots & U^r_{i,n_i} \end{bmatrix},$$
$$U^r_{i,j}(s) := \begin{bmatrix} \frac{1}{s} e^r_{i,j,1} & e^r_{i,j,2} \end{bmatrix},$$

$$U^\rho := \begin{bmatrix} U^\rho_1 & U^\rho_2 & \cdots & U^\rho_n \end{bmatrix},$$
$$U^\rho_i := \begin{bmatrix} U^\rho_{1,i} & U^\rho_{2,i} & \cdots & U^\rho_{i-1,i} & U^\rho_{i+1,i} & \cdots & U^\rho_{n,i} \end{bmatrix},$$

$$U^\rho_{i,j}(s) := \begin{bmatrix} \frac{1}{s} e^\rho_{i,j,1} & e^\rho_{i,j,2} \end{bmatrix}, \quad U^{\rho b} := \begin{bmatrix} U^{\rho b}_1 & U^{\rho b}_2 & \cdots & U^{\rho b}_n \end{bmatrix},$$

$$U^{\rho b}_i := \begin{bmatrix} U^{\rho b}_{1,i} & U^{\rho b}_{2,i} & \cdots & U^{\rho b}_{i-1,i} & U^{\rho b}_{i+1,i} & \cdots & U^{\rho b}_{n,i} \end{bmatrix},$$
$$U^{\rho b}_{i,j}(s) := \begin{bmatrix} \frac{1}{s} e^{\rho b}_{i,j,1} & e^{\rho b}_{i,j,2} \end{bmatrix}.$$

By defining $q := [\,q_1 \;\cdots\; q_n\,]^T$, $q_d := [\,q_{d,1} \;\cdots\; q_{d,n}\,]^T$, $c := [\,c_1 \;\cdots\; c_n\,]^T$, and $e := q_d - q$, the overall control system can now be represented as in Figure 2. Furthermore, since the $\mathcal{L}_2$-induced norm of each of $\Delta^r_{i,j,l}$, $\Delta^\rho_{j,i,l}$, and $\Delta^{\rho b}_{i,j,l}$ $(l = 1, 2)$ is less than $(1/\sqrt{2})$, due to its structure given above, the $\mathcal{L}_2$-induced norm of Δ^o_{LTV} is less than 1.

B. *Controller derivation*

To find a solution to the optimisation problem (14), we define a new problem:

$$\inf_{\bar{\hat{K}} \text{ stabilising } \hat{\hat{P}}} \left\| \begin{bmatrix} W_1 \left(I + \hat{\hat{P}}\bar{\hat{K}}\right)^{-1} \\ \xi \bar{\hat{K}} \left(I + \hat{\hat{P}}\bar{\hat{K}}\right)^{-1} \end{bmatrix} \right\|_\infty = \gamma^{\text{opt}} \tag{33}$$

where $\bar{\hat{K}}$ and $\hat{\hat{P}}$ are to be defined below. As will be shown below, if there exists a solution, a stabilising controller, $\hat{K}$, providing γ^{opt}, to the problem (33), then, there exists a solution, a stabilising controller, K, providing γ^{opt}, to the problem given in (14). To show this we first let $\hat{K} := \hat{P}K = [\,\hat{K}^{r^T} \;\; \hat{K}^{\rho^T} \;\; \hat{K}^{\rho b^T}\,]^T$. This leads to $P_o K = \check{P}\hat{K}$, where $\check{P}(s) := (1/s)\bar{P}e^{-Hs} = [\,P^r(s) \;\; P^\rho(s) \;\; P^{\rho b}(s)\,]$, where $P^r(s) := (1/s)\bar{P}_r e^{-H^r s}$, $P^\rho(s) := (\sqrt{2}/s)\bar{P}_\rho e^{-H^\rho s}$ and $P^{\rho b}(s) := (\sqrt{2}/s)\bar{P}_{\rho b} e^{-H^{\rho b} s}$ with $H^r := \text{diag}(h^r)$, $H^\rho := \text{diag}(h^\rho)$, and $H^{\rho b} := \text{diag}(h^{\rho b})$. Note that, P^r and P^ρ are block diagonal. Although $P^{\rho b}$ is not block diagonal, by permuting the columns of $I_{n(n-1)}$, we can find a non-singular matrix T, satisfying $T^T T = TT^T = I$, such that $\hat{P}^{\rho b} := P^{\rho b}T^T$ is block diagonal. Then, by defining $\bar{\hat{K}}^{\rho b} := T\hat{K}^{\rho b}$, $P^{\rho b}\hat{K}^{\rho b} = \hat{P}^{\rho b}\bar{\hat{K}}^{\rho b}$ is obtained and $\hat{\check{P}} := [\,P^r \;\; \hat{P}^\rho \;\; \hat{P}^{\rho b}\,]$, $\bar{\hat{K}} := [\,\hat{K}^{r^T} \;\; \hat{K}^{\rho^T} \;\; \bar{\hat{K}}^{\rho b^T}\,]^T$. Hence, $P_o K = \check{P}\hat{K} = \hat{\check{P}}\bar{\hat{K}}$ implying that $(I + P_o K)^{-1} = (I + \check{P}\hat{K})^{-1} = (I + \hat{\check{P}}\bar{\hat{K}})^{-1}$. Furthermore, due to the fact that $\|\bar{\hat{K}}^{\rho b}\|_\infty = \|T\hat{K}^{\rho b}\|_\infty$, the infimum obtained in both problems (14) and (33) are equal to γ^{opt}. These results show the equivalency of the two problems. Therefore, we first find a solution to the problem given in (33) and then, using this solution, we obtain a solution to the problem given in (14).

It can be shown that problem (33) can be decomposed into the following problems:

$$\inf_{\bar{\hat{K}}_i \text{ stabilising } \hat{\hat{P}}_i} \left\| \begin{bmatrix} W_1(1 + \hat{\hat{P}}_i\bar{\hat{K}}_i)^{-1} \\ \xi \bar{\hat{K}}_i(1 + \hat{\hat{P}}_i\bar{\hat{K}}_i)^{-1} \end{bmatrix} \right\|_\infty =: \gamma_i^{\text{opt}} \tag{34}$$

for $i = 1, \ldots, n$, where $\hat{\hat{P}}_i := [\,P^r_i \;\; P^\rho_i \;\; \hat{P}^{\rho b}_i\,]$, $\bar{\hat{K}}_i := [\,\hat{K}^{r^T}_i \;\; \hat{K}^{\rho^T}_i \;\; \bar{\hat{K}}^{\rho b^T}_i\,]^T$, where P^r_i, P^ρ_i, $\hat{P}^{\rho b}_i$, $\hat{K}^r_i$, $\hat{K}^\rho_i$, and $\bar{\hat{K}}^{\rho b}_i$ are the ith diagonal blocks of P^r, P^ρ, $\hat{P}^{\rho b}$, $\hat{K}^r$, $\hat{K}^\rho$, and $\bar{\hat{K}}^{\rho b}$, respectively. Note that $\gamma^{\text{opt}} = \max_i(\gamma_i^{\text{opt}})$, which means that an optimal (respectively, suboptimal) solution to (33) is obtained by combining the optimal (respectively, suboptimal) solutions to the problems in (34).

The problems defined in (34) are similar to the problem considered in Quet et al. (2002). Therefore, as it was done in Quet et al. (2002), we will decompose the problems (34) into subproblems involving single delays and find a suboptimal solution to each problem in (34). For this, as in Quet et al. (2002), let us consider the coprime factorisations of $\check{P}_i$ in $\mathcal{H}^\infty$:

$$\hat{\hat{P}}_i(s) = \begin{bmatrix} P^r_i(s) & P^\rho_i(s) & \hat{P}^{\rho b}_i(s) \end{bmatrix} = N_i(s)M_i^{-1}(s) = \tilde{M}_i^{-1}(s)\tilde{N}_i(s), \tag{35}$$

where

$$N_i(s) = \tilde{N}_i(s) = \frac{1}{s+\epsilon}\begin{bmatrix} e^{-h^r_{i,1}s} & \cdots & e^{-h^r_{i,n_i}s} & \sqrt{2}e^{-h^\rho_{1,i}s} \\ \cdots & \sqrt{2}e^{-h^\rho_{n,i}s} & \sqrt{2}e^{-h^{\rho b}_{i,1}s} \cdots & \sqrt{2}e^{-h^{\rho b}_{i,n}s} \end{bmatrix},$$

and

$$\tilde{M}_i(s) = \frac{s}{s+\epsilon}, \quad M_i(s) = \frac{s}{s+\epsilon} I_{n_i+2n(n-1)},$$

where $\epsilon > 0$ is arbitrary.

Now, a parametrisation of all controllers $\bar{\hat{K}}_i(s)$ which stabilise $\hat{\tilde{P}}_i(s)$ can be obtained as

$$\bar{\hat{K}}_i(s) = [X_i(s) + M_i(s)Q_i(s)][Y_i(s) - N_i(s)Q_i(s)]^{-1} \tag{36}$$

in terms of $Q_i \in \mathcal{H}^\infty$, where $Q_i := [\,Q_i^{rT} \;\; Q_i^{\rho T} \;\; Q_i^{\rho b\,T}]^T$, where $Q_i^r := [\,Q_{i,1}^r \;\; \cdots \;\; Q_{i,n_i}^r]^T$, $Q_i^\rho := [\,Q_{1,i}^\rho \;\; \cdots \;\; Q_{i-1,i}^\rho \;\; Q_{i+1,i}^\rho \;\; \cdots \;\; Q_{n,i}^\rho]^T$, $Q_i^{\rho b} := [\,Q_{i,1}^{\rho b} \;\; \cdots \;\; Q_{i,i-1}^{\rho b} \;\; Q_{i,i+1}^{\rho b} \;\; \cdots \;\; Q_{i,n}^{\rho b}]^T$. Furthermore, $X_i \in \mathcal{H}^\infty$ and $Y_i \in \mathcal{H}^\infty$ satisfy the Bezout identity: $\tilde{M}_i(s)Y_i(s) + \tilde{N}_i(s)X_i(s) = 1$. Since $\lim_{s\to 0} \tilde{M}_i(s) = 0$, to satisfy the Bezout identity we must have $\lim_{s\to 0} \tilde{N}_i(s)\;\; X_i(s) = 1$, or equivalently, $\epsilon^{-1}[1 \cdots 1\ \sqrt{2} \cdots \sqrt{2}\ \sqrt{2} \cdots \sqrt{2}]X_i(0) = 1$. Thus, we choose $X_i(s) = [\alpha_{i,1}^r \;\; \ldots \;\; \alpha_{i,n_i}^r \;\; (\alpha_{1,i}^\rho/\sqrt{2}) \;\; \ldots \;\; (\alpha_{i-1,i}^\rho/\sqrt{2})\;(\alpha_{i+1,i}^\rho/\sqrt{2}) \ldots (\alpha_{n,i}^\rho/\sqrt{2})\;(\alpha_{i,1}^{\rho b}/\sqrt{2}) \ldots (\alpha_{i,i-1}^{\rho b}/\sqrt{2})\;(\alpha_{i,i+1}^{\rho b}/\sqrt{2}) \ldots (\alpha_{i,n}^{\rho b}/\sqrt{2})]^T\epsilon$, where positive numbers $\alpha_{i,l}^\bullet$ satisfy (18). Thus, $Y_i(s)$ must be chosen as follows:

$$\begin{aligned} Y_i(s) &= \tilde{M}_i^{-1}(s)\big[1 - \tilde{N}_i(s)X_i(s)\big] \\ &= \frac{s+\epsilon}{s} - \frac{\epsilon}{s}\left[\sum_{j=1}^{n_i} \alpha_{i,j}^r e^{-h_{i,j}^r s} + \sum_{j=1,j\neq i}^{n} \alpha_{j,i}^\rho e^{-h_{j,i}^\rho s} + \sum_{j=1,j\neq i}^{n} \alpha_{i,j}^{\rho b} e^{-h_{i,j}^{\rho b} s}\right]. \end{aligned}$$

Then, substituting (36) into (34) and re-arranging terms, we obtain,

$$\inf_{Q_i\in\mathcal{H}^\infty} \left\| \begin{bmatrix} W_1(s)\dfrac{s}{s+\epsilon}\left[\dfrac{s+\epsilon}{s} - \dfrac{\epsilon}{s}\left(\displaystyle\sum_{j=1}^{n_i} \alpha_{i,j}^r e^{-h_{i,j}^r s} + \sum_{j=1,j\neq i}^{n} \alpha_{j,i}^\rho e^{-h_{j,i}^\rho s} + \sum_{j=1,j\neq i}^{n} \alpha_{i,j}^{\rho b} e^{-h_{i,j}^{\rho b} s}\right)\right. \\ \left. -\dfrac{1}{s+\epsilon}\left(\displaystyle\sum_{j=1}^{n_i} e^{-h_{i,j}^r s} Q_{i,j}^r(s) + \sqrt{2}\sum_{j=1,j\neq i}^{n} e^{-h_{j,i}^\rho s} Q_{j,i}^\rho(s) + \sqrt{2}\sum_{j=1,j\neq i}^{n} e^{-h_{i,j}^{\rho b} s} Q_{i,j}^{\rho b}(s)\right)\right] \\ \xi(s)\dfrac{s}{s+\epsilon}\left[\displaystyle\sum_{j=1}^{n_i}\left(\alpha_{i,j}^r\epsilon + \frac{s}{s+\epsilon}Q_{i,j}^r(s)\right)\frac{s}{s+\epsilon} + \sum_{j=1,j\neq i}^{n}\left(\frac{\alpha_{j,i}^\rho}{\sqrt{2}}\epsilon + \frac{s}{s+\epsilon}Q_{j,i}^\rho(s)\right)\frac{s}{s+\epsilon}\right. \\ \left. + \displaystyle\sum_{j=1,j\neq i}^{n}\left(\frac{\alpha_{i,j}^{\rho b}}{\sqrt{2}}\epsilon + \frac{s}{s+\epsilon}Q_{i,j}^{\rho b}(s)\right)\frac{s}{s+\epsilon}\right] \end{bmatrix} \right\|_\infty := \gamma_i^{opt}$$

Using (18), we can write the following,

$$\begin{aligned} &\sum_{j=1}^{n_i} \inf_{Q_{i,j}^r\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{i,j}^r W_1(s)\Big[Y_{i,j}^r(s) - N_{i,j}^r(s)Q_{i,j}^r(s)\Big]M_{i,j}^r(s) \\ \xi(s)\Big[X_{i,j}^r(s) + M_{i,j}^r(s)Q_{i,j}^r(s)\Big]M_{i,j}^r(s) \end{bmatrix} \right\|_\infty \\ &+ \sum_{j=1,j\neq i}^{n} \inf_{Q_{j,i}^\rho\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{j,i}^\rho W_1(s)\Big[Y_{j,i}^\rho(s) - N_{j,i}^\rho(s)Q_{j,i}^\rho(s)\Big]M_{j,i}^\rho(s) \\ \xi(s)\Big[X_{j,i}^\rho(s) + M_{j,i}^\rho(s)Q_{j,i}^\rho(s)\Big]M_{j,i}^\rho(s) \end{bmatrix} \right\|_\infty \\ &+ \sum_{j=1,j\neq i}^{n} \inf_{Q_{i,j}^{\rho b}\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{i,j}^{\rho b} W_1(s)\Big[Y_{i,j}^{\rho b}(s) - N_{i,j}^{\rho b}(s)Q_{i,j}^{\rho b}(s)\Big]M_{i,j}^{\rho b}(s) \\ \xi(s)\Big[X_{i,j}^{\rho b}(s) + M_{i,j}^{\rho b}(s)Q_{i,j}^{\rho b}(s)\Big]M_{i,j}^{\rho b}(s) \end{bmatrix} \right\|_\infty \\ &=: \gamma_i \geq \gamma_i^{\text{opt}} \end{aligned}$$

where

$$\begin{aligned} Y_{i,j}^r(s) &:= \frac{s+\epsilon}{s} - \frac{\epsilon}{s}e^{-h_{i,j}^r s}, \quad & X_{i,j}^r(s) &:= \alpha_{i,j}^r\epsilon, \\ N_{i,j}^r(s) &:= \frac{1}{\alpha_{i,j}^r(s+\epsilon)}e^{-h_{i,j}^r s}, \quad & M_{i,j}^r(s) &:= \frac{s}{s+\epsilon}, \\ Y_{i,j}^\bullet(s) &:= \frac{s+\epsilon}{s} - \frac{\epsilon}{s}e^{-h_{i,j}^\bullet s}, \quad & X_{i,j}^\bullet(s) &:= \frac{\alpha_{i,j}^\bullet}{\sqrt{2}}\epsilon, \\ N_{i,j}^\bullet(s) &:= \frac{\sqrt{2}}{\alpha_{i,j}^\bullet(s+\epsilon)}e^{-h_{i,j}^\bullet s}, \quad & M_{i,j}^\bullet(s) &:= \frac{s}{s+\epsilon} \end{aligned}$$

where the superscript $\bullet$ represents ρ or ρb. Therefore, we define the following problems, each of which involves a single delay:

$$\inf_{Q_{i,j}^r\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{i,j}^r W_1\Big[Y_{i,j}^r - N_{i,j}^r Q_{i,j}^r\Big]M_{i,j}^r \\ \xi\Big[X_{i,j}^r + M_{i,j}^r Q_{i,j}^r\Big]M_{i,j}^r \end{bmatrix} \right\|_\infty =: \gamma_{i,j}^r \tag{37}$$

$$\inf_{Q_{j,i}^\rho\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{j,i}^\rho W_1\Big[Y_{j,i}^\rho - N_{j,i}^\rho Q_{j,i}^\rho\Big]M_{j,i}^\rho \\ \xi\Big[X_{j,i}^\rho - M_{j,i}^\rho Q_{j,i}^\rho\Big]M_{j,i}^\rho \end{bmatrix} \right\|_\infty =: \gamma_{j,i}^\rho \tag{38}$$

$$\inf_{Q_{i,j}^{\rho b}\in\mathcal{H}^\infty} \left\| \begin{bmatrix} \alpha_{i,j}^{\rho b} W_1\Big[Y_{i,j}^{\rho b} - N_{i,j}^{\rho b} Q_{i,j}^{\rho b}\Big]M_{i,j}^{\rho b} \\ \xi\Big[X_{i,j}^{\rho b} - M_{i,j}^{\rho b} Q_{i,j}^{\rho b}\Big]M_{i,j}^{\rho b} \end{bmatrix} \right\|_\infty =: \gamma_{i,j}^{\rho b} \tag{39}$$

where $j = 1, \ldots, n_i$ for the problem defined in (37) and $j = 1, \ldots, n,\ j \neq i$ for the problems defined in (38) and (39). Note that, a suboptimal solution to (34) can be obtained by combining optimal solutions of (37)–(39), since

$$\gamma_i^{\text{opt}} \leq \sum_{j=1}^{n_i} \gamma_{i,j}^r + \sum_{j=1,j\neq i}^{n} \gamma_{j,i}^\rho + \sum_{j=1,j\neq i}^{n} \gamma_{i,j}^{\rho b} =: \tilde{\gamma}_i\,. \tag{40}$$

To solve the problems in (37)–(39) using the results of Quet et al. (2002), let us define $C_{i,j}^\bullet(s) := [X_{i,j}^\bullet(s) + M_{i,j}^\bullet(s)Q_{i,j}^\bullet(s)][Y_{i,j}^\bullet(s) - N_{i,j}^\bullet(s)Q_{i,j}^\bullet(s)]^{-1}$, where the superscript

• represents r, ρ or ρb. Then, we can write

$$Q^{\bullet}_{i,j}(s) = \left[M^{\bullet}_{i,j}(s) + C^{\bullet}_{i,j}(s)N^{\bullet}_{i,j}(s)\right]^{-1}\left[C^{\bullet}_{i,j}(s)Y^{\bullet}_{i,j}(s) - X^{\bullet}_{i,j}(s)\right]. \tag{41}$$

If (41) is substituted into (37)–(39), the optimal controller for each of these problems can be obtained as (Toker and Özbay 1995)

$$C^{\bullet}_{i,j}(s) = \alpha^{\bullet}_{i,j}\bar{\gamma}^{\bullet}_{i,j}\left(\frac{sh^{\bullet}_{i,j} - k^{\bullet}_{i,j}}{sh^{\bullet}_{i,j}}\right)\frac{1}{1 + F^{\bullet}_{i,j}(sh^{\bullet}_{i,j})} \tag{42}$$

where $F^{\bullet}_{i,j}$ is a finite impulse response filter and $k^{\bullet}_{i,j}$ and $\bar{\gamma}^{\bullet}_{i,j}$ are constants to be calculated as in Quet et al. (2002). Then, using (41), $Q^{\bullet}_{i,j}$'s are found and substituted back to obtain Q_i. Once Q_i is found, we obtain $\hat{K}_i := [\,\hat{K}_i^{r\,T} \quad \hat{K}_i^{\rho\,T} \quad \hat{K}_i^{\rho b\,T}\,]^T$ from (36), which defines $\hat{K}$. Then, a controller which is a suboptimal solution to the problem given in (14) can be obtained as $K = P^{\dagger}\hat{K}$, where $P^{\dagger}$ is a left inverse of $\hat{P}$. Using

$$P^{\dagger} = \begin{bmatrix} I_m & 0 & 0 \\ 0 & \sqrt{2}I_{n(n-1)} & 0 \end{bmatrix},$$

the controller given in (15) is obtained.

C. *Steady-state rates*

The tracking error can be written as $e(s) = (I + P_o(s)K(s))^{-1}(1/s)[q_d + c(s)]$. Substituting this into (20), $u_\infty := \lim_{t\to\infty} u(t)$ can be obtained as

$$\begin{aligned} u_\infty &= \lim_{s\to 0} su(s) = \lim_{s\to 0} K(s)(I + P_o(s)K(s))^{-1}(q_d + c(s)) \\ &= \lim_{s\to 0} K_o(s)\left(s^2 I + \bar{P}_o(s)K_o(s)\right)^{-1}(sq_d + sc(s)), \end{aligned} \tag{43}$$

where $\bar{P}_o(s) := sP_o(s)$ and $K_o(s) := sK(s)$. Obtaining $K(s)$ from (15) and using (16), (17), and (42), we obtain

$$\bar{\kappa} := \lim_{s\to 0} K_o(s) = \begin{bmatrix} \bar{\kappa}^r \\ \bar{\kappa}^\rho \end{bmatrix},$$

where

$$\bar{\kappa}^r := \begin{bmatrix} \bar{\kappa}^r_{11} & & \\ \vdots & 0 & \\ \bar{\kappa}^r_{1n_1} & & \\ & \ddots & \\ & & \bar{\kappa}^r_{n1} \\ & 0 & \vdots \\ & & \bar{\kappa}^r_{nn_n} \end{bmatrix}$$

and

$$\bar{\kappa}^\rho := \begin{bmatrix} \bar{\kappa}^\rho_{21} & & \\ \vdots & 0 & \\ \bar{\kappa}^\rho_{n1} & & \\ & \ddots & \\ & & \bar{\kappa}^\rho_{1n} \\ & 0 & \vdots \\ & & \bar{\kappa}^\rho_{(n-1)n} \end{bmatrix},$$

and

$$\begin{aligned} G &:= \lim_{s\to 0} \bar{P}_o(s)K_o(s) \\ &= \begin{bmatrix} \sum_{j=1}^{n_1}\bar{\kappa}^r_{1,j} + \sum_{j=2}^{n}\bar{\kappa}^\rho_{j,1} & -\bar{\kappa}^\rho_{1,2} & \cdots & -\bar{\kappa}^\rho_{1,n} \\ -\bar{\kappa}^\rho_{2,1} & \sum_{j=1}^{n_2}\bar{\kappa}^r_{2,j} + \sum_{j=1,j\neq 2}^{n}\bar{\kappa}^\rho_{j,2} & \cdots & -\bar{\kappa}^\rho_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ -\bar{\kappa}^\rho_{n,1} & -\bar{\kappa}^\rho_{n,2} & \cdots & \sum_{j=1}^{n_n}\bar{\kappa}^r_{n,j} + \sum_{j=1}^{n-1}\bar{\kappa}^\rho_{j,n} \end{bmatrix}, \end{aligned}$$

where

$$\bar{\kappa}^r_{i,j} = -\frac{\alpha^r_{i,j}}{\sum_{k=1}^{n_i}\frac{\alpha^r_{i,k}h^r_{i,k}}{\bar{\gamma}^r_{i,k}} + \sum_{k=1,k\neq i}^{n}\frac{\alpha^\rho_{k,i}h^\rho_{k,i}}{\bar{\gamma}^\rho_{k,i}} + \sum_{k=1,k\neq i}^{n}\frac{\alpha^{\rho b}_{i,k}h^{\rho b}_{i,k}}{\bar{\gamma}^{\rho b}_{i,k}}}$$

and

$$\bar{\kappa}^\rho_{j,i} = -\frac{\alpha^\rho_{j,i}}{\sum_{k=1}^{n_i}\frac{\alpha^r_{i,k}h^r_{i,k}}{\bar{\gamma}^r_{i,k}} + \sum_{k=1,k\neq i}^{n}\frac{\alpha^\rho_{k,i}h^\rho_{k,i}}{\bar{\gamma}^\rho_{k,i}} + \sum_{k=1,k\neq i}^{n}\frac{\alpha^{\rho b}_{i,k}h^{\rho b}_{i,k}}{\bar{\gamma}^{\rho b}_{i,k}}}.$$

Then, the limit given in (43) is obtained as

$$u_\infty = \bar{\kappa}G^{-1}c_\infty, \tag{44}$$

where $c_\infty := [\,c_{1,\infty} \quad \ldots \quad c_{n,\infty}\,]^T = \lim_{t\to\infty} c(t)$.